

**PSAIA**  
**(Protein Structure and Interaction Analyzer)**  
**User Manual**

PSAIA is a software tool used for polypeptide geometric structure analysis and determination of residue interaction pairs and binding residues.

PSAIA is divided in to modules:

- a) **geometric structure analysis module** – calculation of various attributes of polypeptide structure: *Solvent Accessible Surface Area (ASA)*, *Relative ASA* , *Depth Index (DPX)* and *Protrusion Index (CX)*
- b) **interaction analysis module** – identification of interaction pairs and binding residues using one of four implemented algorithms: *Atom Core Distance*, *Van der Waals Radii Distance*, *ASA Change After Complexsation* and *PIADA algorithm*

Both modules use standard *Protein Data Bank* files as input data, and offer a choice of *XML* (.xml) and/or *table* (.tbl) output file format.

# 1 Installation

PSAIA can be used both from GUI and console. Also, there are versions for MS Windows and Linux.

## 1.1 MS Windows installation

For installation software on MS Windows just use *PSAIA-1.0-windows-installer.exe* and follow instruction. Application parameters files will be installed in *amac\_data* folder. In the installation folder there are 3 executable files:

- *psaia.exe*
- *psa.exe*
- *pia.exe*,

and the *aa\_contacts.txt* file that contains contacts pairs for PSAIA and PIA applications (note: if this file is removed the applications will report an error and stop their work).

*Psaia.exe* is GUI application, while *psa.exe* and *pia.exe* are console applications.

## 1.2 Linux installation

For installation software on Linux just use *PSAIA-1.0-linux-installer.bin* and follow instruction. Before starting program it is necessary to change permissions with *chmod 744 PSAIA-1.0-linux-installer.bin*. Start application with *./PSAIA-1.0-linux-installer.bin*. Application parameters files will be installed in *amac\_data* folder. In the installation folder there are 3 executable files:

- *psaia*
- *psa*
- *pia*

and the *aa\_contacts.txt* file that contains contacts pairs for PSAIA and PIA applications (note: if this file is removed the applications will report an error and stop their work).

Psaia is GUI application, while psa and pia are console applications. For starting applications use *pia.sh*, *psa.sh* and *psaia.sh* shell scripts. Before starting applications change permissions with *chmod 744 \*.sh*.

There is also a source code in packed in PSAIA-1.0.tar.gz. If you want to install that way you should have preinstall Qt 4.1.4 libraries. First step of installation is going to *\$CURRENT\_DIRECTORY/psaia/make/linux/*. There are three directories. One for each of the applications:

- *psaia*
- *psa*
- *pia*

In each directory there are QMAKE .pro (i.e. psa.pro, pia.pro and psaia.pro) files that create a *Makefile*. Running 'qmake' in one of the make directories creates a *Makefile*. By default 'make' will install the package's files in *\$CURRENT\_DIRECTORY/psaia/bin/linux/*. Application parameters files are in *\$CURRENT\_DIRECTORY/psaia/bin/amac\_data*.

The user can change the output path where an application is to be built in and/or the input paths of the source files.

The *aa\_contacts.txt* file that contains contacts pairs for PSAIA and PIA applications has to be located in the directory in which the application binaries are built (note: if this file is removed the applications will report an error and stop their work).

The example of the .pro file is as follows:

```
TEMPLATE = app
LANGUAGE = C++

INCLUDEPATH += ../../../../protein

HEADERS = ../../../../psaia_app/psaia.h\
          ../../../../psaia_app/crunpsa.h\
          ../../../../psaia_app/run_pia.h\
          ../../../../psaia_app/status_message.h\
          ../../../../protein/addition.h\
          ../../../../protein/atom_grid.h\
          ../../../../protein/atom_space_calculations.h\
          ../../../../protein/chain_residue.h\
          ../../../../protein/circles.h\
          ../../../../protein/pdb_peptide.h\
          ../../../../protein/protein_chain.h\
```

```

        ../../../../protein/residue_atom.h\
        ../../../../protein/residue_contact.h\
        ../../../../protein/string_operation.h\
        ../../../../protein/xml_writer.h

SOURCES = ../../../../psaia_app/main.cpp\
          ../../../../psaia_app/psaia_linux.cpp\
          ../../../../psaia_app/crunpsa_linux.cpp\
          ../../../../psaia_app/run_pia_linux.cpp\
          ../../../../psaia_app/status_message.cpp\
          ../../../../protein/addition.cpp\
          ../../../../protein/atom_grid.cpp\
          ../../../../protein/atom_space_calculations.cpp\
          ../../../../protein/chain_residue.cpp\
          ../../../../protein/circles.cpp\
          ../../../../protein/pdb_peptide.cpp\
          ../../../../protein/protein_chain.cpp\
          ../../../../protein/residue_atom.cpp\
          ../../../../protein/residue_contact.cpp\
          ../../../../protein/string_operation.cpp\
          ../../../../protein/xml_writer.cpp

FORMS = ../../../../psaia_app/psaia.ui

DESTDIR = ../../../../bin/linux/psaia

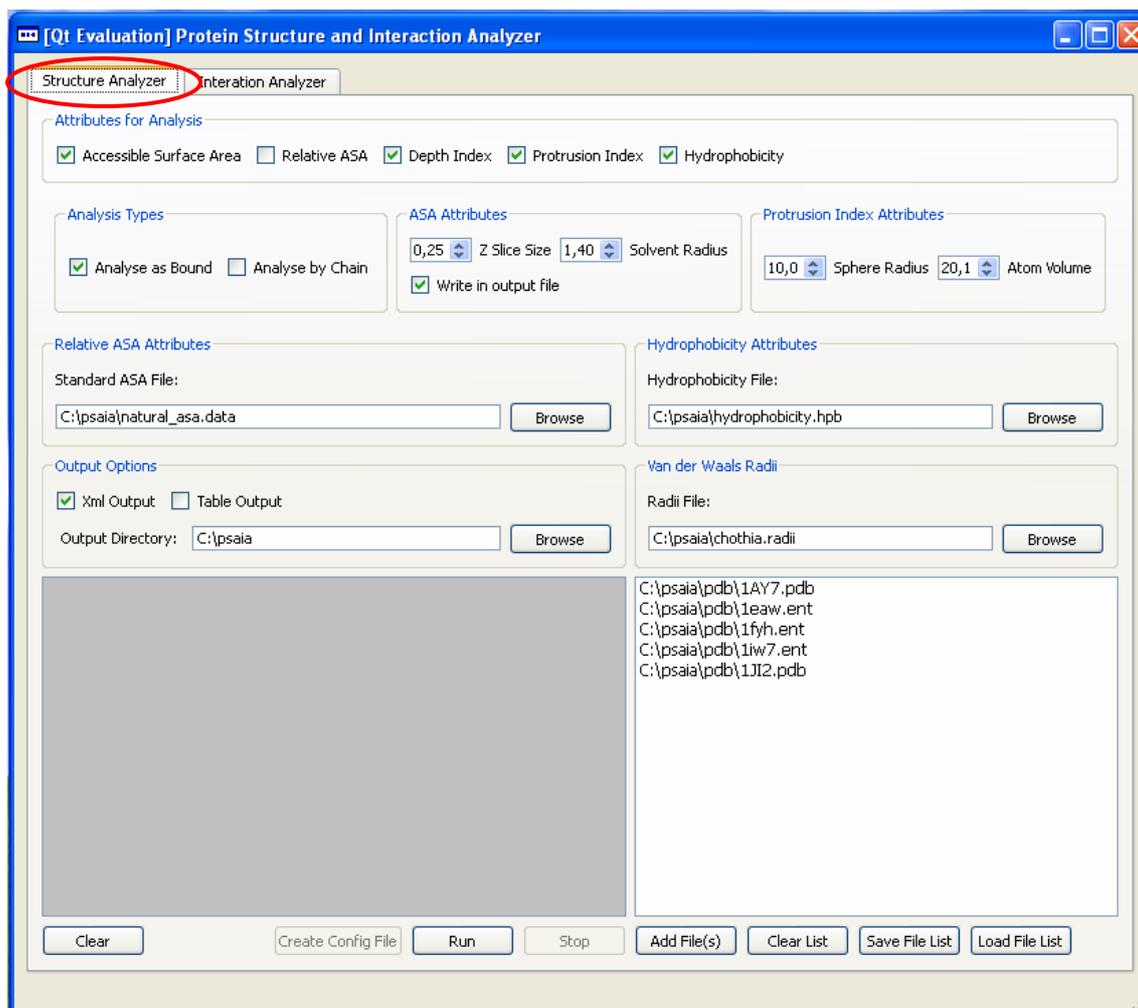
CONFIG += release warn_on_thread qt no_lflags_merge x11 stl
QT += core gui qt3support

```

For starting that way compiled application you don't need additional shell scripts, so you can start it using pia, psa and psaia binaries.

## 2 Geometric structure analysis module (Structure analyzer)

*Structure analyzer* module can be selected by clicking on the “Structure Analyzer” tab in the main window. This module is used for analysis of one or more geometric attributes of polypeptides. Structure analysis can be performed on the complete structure in a PDB file or on each chain individually.



**Figure 1: Structure Analyzer module**

The module's main window is made out of multiple groups: attribute groups used for module work adjustment, module work control group and input file control group.

## 2.1 Module work adjustment

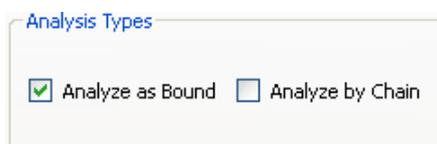
**Residue attributes for analysis** - *Accessible Surface Area, Relative ASA, Depth Index, Protrusion Index, and Hydrophobicity*. The user can choose an arbitrary number (1 to 5) of attributes to be calculated on loaded files.

*Remark:* calculation of Relative ASA and/or Depth Index requires calculation of ASA. In this situation it is not required of the user to switch on ASA attribute since it will be automatically turned on.



**Figure 2: structure attributes for analysis**

The user has a choice of two types of peptide analysis - analysis can be on the complete structure in a PDB file (*Analyze as Bound*) or on each chain within the structure individually (*Analyze by Chain*). Choosing both types of analysis will produce two separate output files.



Analysis Types

Analyze as Bound  Analyze by Chain

**Figure 3: structure analysis types**

**ASA attributes** – probe sphere slicing resolution (*Z\_Slice\_Size*), probe sphere radius (*Probe Radius*). The user is free to exclude ASA calculation results from the output file (it can be used if Relative ASA and/or Depth Index calculation is chosen and the ASA results are of no interest) – (*Write in output file*).



ASA Attributes

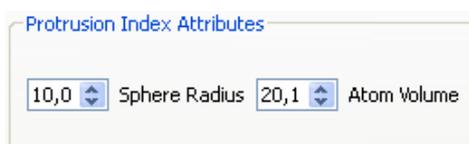
0,25 Z\_Slice Size 1,40 Probe Radius

Write in output file

**Figure 4: ASA Attributes**

There are no specific attributes for *Depth Index* calculation; it is only required to adjust the desired ASA attributes.

**Protrusion Index Attributes** – neighborhood radius (*Sphere Radius*), mean atom volume (*Atom Volume*).



Protrusion Index Attributes

10,0 Sphere Radius 20,1 Atom Volume

**Figure 5: Protrusion Index Attributes**

**Relative ASA** calculation requires a standard ASA value file (file that contains standard ASA values for amino acids). Software package includes one file (*natural\_asa.asa*) containing standard ASA values calculated using *Z\_Slice\_Value* of 0,25 Å and *Solvent Radius* of 1,40 Å. The user is free to use any other properly formatted file.

The file containing standard ASA values has to be formatted in the following manner. The file has to contain 6 columns separated by a white space or a tab and these columns have to contain the values as described in the following table:

**Table 1: standard ASA file format**

Column no.	Description	Type	Maximum length
1	residue name	String	3
2	total ASA	Double	5
3	main chain ASA	Double	5
4	side chain ASA	Double	5
5	non-polar ASA	Double	5
6	polar ASA	Double	5

Comments in the file should be defined with a starting '#' in the line. All NON EMPTY wrongly formatted or non-commented lines will be interpreted as an error and the application will stop its work.

Example of standard ASA notation:

**ALA 107.24 43.32 63.92 76.06 31.17**

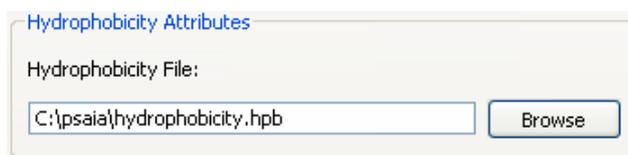


**Figure 6: Relative ASA Attributes**

**Hydrophobicity attributes** – input file containing amino acid hydrophobicity values. Software package includes one file (hydrophobicity.hpb) containing hydrophobicity values. The user is free to use any other properly formatted file.

The file containing hydrophobicity values has to be formatted in the following manner. The hydrophobicity notation starts with a one to three letter amino acid name abbreviation followed by white space or tab separated hydrophobicity double type value.

Comments in the file should be defined with a starting '#' in the line. All NON EMPTY wrongly formatted or non-commented lines will be interpreted as an error and the application will stop its work.



**Figure 7: Hydrophobicity Attributes**

**Van der Waals radii** file has to be supplied for structure analysis. Radii values are read from radii files. Software package includes one file (chothia.radii) containing radii values according to Chothia. The user is free to use any other properly formatted file.

The file containing radii values has to be formatted in the following manner.

Each radii notation starts with a zero to three letter amino acid name abbreviation and a one to four letter long atom name; the amino acid name and the atom name have to be connected with an underscore ( \_ ), and can be up to eight characters long. Following the amino acid and atom designator is a double type radii value.

Comments in the file should be defined with a starting '#' in the line. All NON EMPTY wrongly formatted or non-commented lines will be interpreted as an error and the application will stop its work.

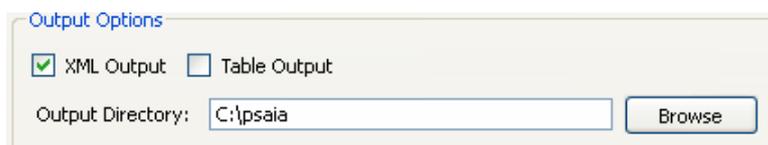
Van der Waals radii example for *Alanine*:

```
ALA_N 1.65
ALA_CA 1.87
ALA_C 1.76
ALA_O 1.40
ALA_CB 1.87
```



**Figure 8: Van der Waals Radii file input group**

**Output options** – XML (*XML Output*) and Table (*Table Output*) output format; folder in which the output files are to be stored (*Output Directory*).



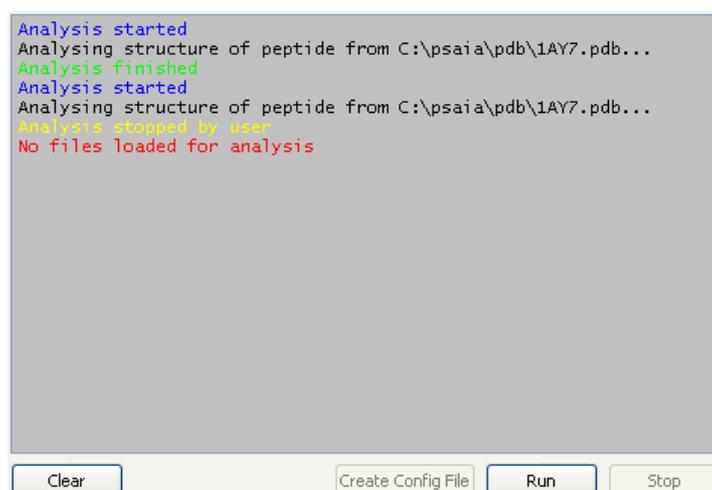
**Figure 9: Structure Analyzer Output Options**

Output file names are constructed out of the PDB identifier of the analyzed peptide (ex. 1AY7) and of the analysis type id connected with an underscore (**bound** for complete structure analysis and **unbound** for chain analysis). There are four different output file formats possible, for example peptide 1ay7 can have next four output files: **1ay7\_bound.xml**, **1ay7\_unbound.xml**, **1ay7\_bound.tbl**, **1ay7\_unbound.tbl**.

## 2.2 Module work control

Module work control consists of program flow screen and of *Run*, *Stop*, *Clear* and *Create Config File* buttons.

By clicking the *Create Config File* button user can create a configuration file that can be used for use with the console version of this module.



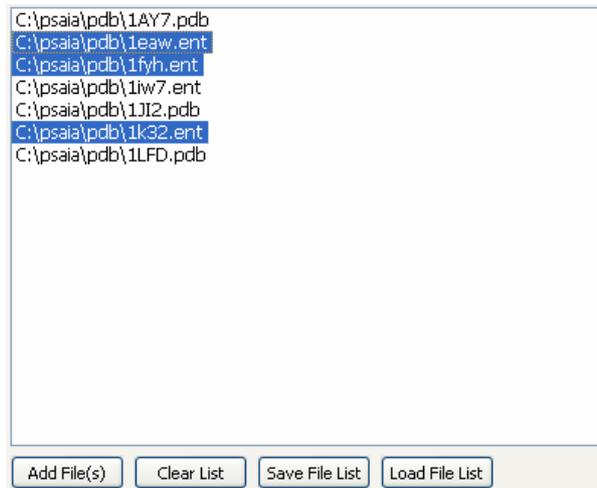
**Figure 10: module work control window**

## 2.3 Input file control

Input file control consists of input file list screen and of *Add File(s)*, *Clear List*, *Save File List* and *Load File List* buttons.

One or more PDB files can be added to the file list by clicking the *Add File(s)* button. Added files can be deleted from the list by clicking the *Clear List* button (all the files in the list will be deleted), or by clicking on file names

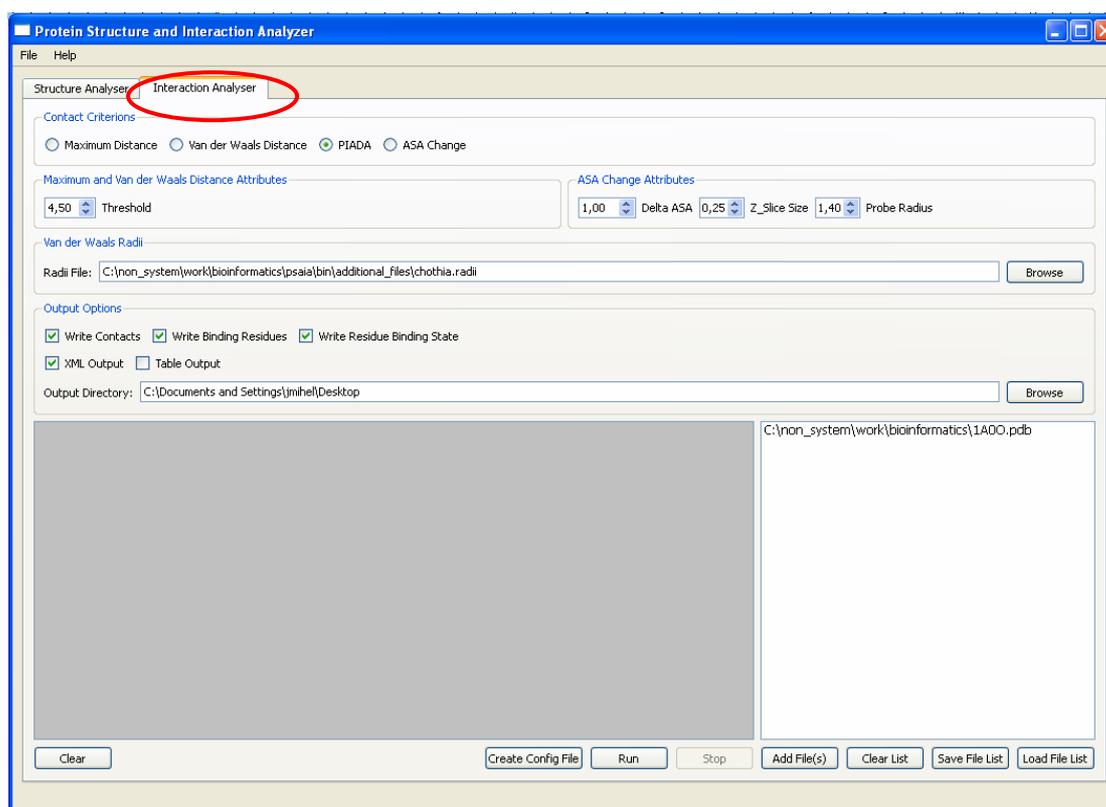
in the file list screen and then pressing *delete* on the keyboard. File lists can be saved by clicking on the *Save File List* button and loaded by clicking on the *Load File List* button.



**Figure 11: input file control window**

### 3 Interaction Analyzer module

*Interaction Analyzer* module can be selected by clicking on the “Interaction Analyzer” tab in the main window. This module is used for determination of interaction pairs between interacting peptide chains and of binding residues of individual chains. The user can choose one out of three algorithms for interaction pairs and binding residues determination or one algorithm used for binding residues determination only.



**Figure 12: Interaction Analyzer module**

The module’s main window is made of multiple parts: attribute groups used for module work adjustment, module work control group and input file control group.

#### 3.1 Module work adjustment

The module can be used for determination of residue interaction pairs and/or binding residues using one of four criteria (*Contact Criteria*): *Maximum Distance*, *Van der Waals Distance*, *PIADA* or *ASA Change*.



**Figure 13: Contact Criteria**

Desired *threshold* used in interaction analysis using *Maximum Distance* or *Van der Waals Distance* can be adjusted in the **Maximum and Van der Waals Distance Attributes**.



**Figure 14: Maximum and Wan der Waals Distance Attributes**

**ASA Change Attributes** – *Delta ASA* value used as the binding residue determination criterion and ASA calculation attributes (*Z\_Slice\_Site* and *Probe Radius*) needed for calculating the Accessible Surface Area before and after complexation.

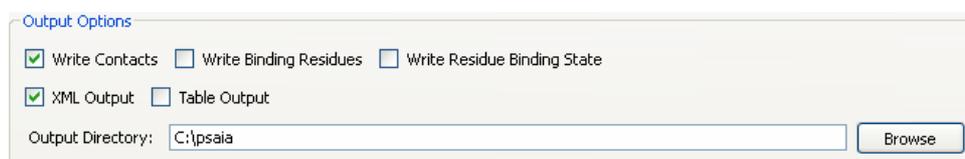


**Figure 15: ASA Change Attributes**

The procedure for Van der Waals radii setting is consistent with the one described in chapter 1.1.

**Output options** – XML (*XML Output*) and Table (*Table Output*) output format; folder in which the output files are to be stored (*Output Directory*). There are three different types of data that can be written in output files:

- a) *Write Contacts* – interaction pairs (residue contacts) are written into the output file
- b) *Write Binding Residues* – residues involved in interactions are written into the output file
- c) *Write Residue Binding State* – true or false value is written into the output file for all residues depending on their interaction involvement



**Figure 16: Interaction Analyzer Output Options**

Output file names are constructed out of the PDB identifier of the analyzed peptide (ex. 1AY7) and of the output data type identifier connected with an underscore (**contacts** for residue contacts output, **binding\_residues** for binding residues output and **binding\_status** for residue binding status). There are six different output file formats possible, for example peptide 1ay7 can have next six output files: **1ay7\_contacts.xml**, **1ay7\_binding\_residues.xml**, **1ay7\_binding\_status.xml**, **1ay7\_contacts.tbl**, **1ay7\_binding\_residues.tbl**, **1ay7\_binding\_status.tbl**.

### **3.2 Module work control**

Module work control is equivalent to the one described in chapter 2.2.

### **3.3 Input file control**

Input file control is equivalent to the one described in chapter 2.3.

## **4 PSA & PIA console modules**

PSA and PIA are console versions of PSAIA's modules Structure Analyzer (PSA – *Protein Structure Analyzer*) and Interaction Analyzer (PIA – *Protein Interaction Analyzer*).

Both modules are run from console by starting `psa.exe` or `pia.exe` (or `pia` or `psa` in Linux environment) with two arguments. The first argument is the path to a configuration file (chapters 3.1 and 3.2) and the second the path to a file-list file (chapter 3.3). For example:

```
psa.exe config_file.txt pdb_list.flis
```

### **4.1 PSA configuration file**

PSA configuration is formatted as follows:

```
analyze_bound: 1
analyze_unbound: 1
calc_asa: 1
z_slice: 0.5
r_solvent: 1.4
write_asa: 1
calc_rasa: 1
standard_asa: C:\psaia\natural_asa.asa
calc_dpx: 1
calc_cx: 1
cx_threshold: 10
```

```

cx_volume: 20.1
calc_hydro: 1
hydro_file: C:\psaia\hydrophobicity.hpb
radii_filename: C:\psaia\chothia.radii
write_xml: 1
write_table: 1
output_dir: C:\psaia\out

```

Each attribute entry contains its name and value separated by a “:”. All attribute names are to be written in small caps and exactly as they are in the following table. All attribute entries have to be written in a new line with nothing preceding the attributes name. The easiest way to create this file is using GUI as described in chapter 2.2.

Comments in the file should be defined with a starting ‘#’ in the line. All NON EMPTY wrongly formatted or non-commented lines will be interpreted as an error and the application will stop its work. Properly formatted lines containing UNKNOWN attributes will also be interpreted as an error.

**Table 2: PSA configuration attributes**

<b>attribute name</b>	<b>Attribute description</b>
analyze_bound & analyze_unbound	These two attributes type of analysis to be done. If their values != 0 bound and/or unbound analysis will be performed.
calc_asa, z_slice, r_solvent, write_asa	If calc_asa value != 0 ASA will be calculated → z_slice & r_solvent represent arguments for calculation of ASA (if they are not listed PSA will use default values). If write_asa attribute has value != 0 ASA will be written into the output file, otherwise it will not (this is used because DPX calculation requires ASA calculation).
calc_rasa standard_asa	If calc_rasa value != 0 relative ASA will be calculated → standard_asa represents the path to a file that contains standard ASA values.
calc_dpx	If calc_dpx value != 0 dpx will be calculated.
calc_cx,	If calc_cx value != 0 cx will be calculated →

cx_threshold, cx_volume	cx_threshold and cx_volume represent arguments for calculation of cx (if they are not listed PSA will use default values).
calc_hydro, hydro_filename	If calc_hydro value != 0 hydrophobicity will be calculated → hydro_filename then represents the path to a file that contains hydrophobicity values.
radii_filename	Path to radii file to be used.
write_xml, write_table	If set to value != 0 the result will be written in xml and/or table format.
output_dir	Path to the output directory.

## 4.2 PIA configuration file

PIA configuration is formatted as follows:

```

contact_criterion: 3
delta_asa: 1.0
z_slice: 0.25
r_solvent: 1.4
radii_filename: C:\psaia\chothia.radii
write_contacts: 1
write_binding_residues: 1
write_binding_state: 1
write_xml: 1
write_table: 1
output_dir: C:\psaia\out

```

Each attribute entry contains its name and value separated by a “:”. All attribute names are to be written in small caps and exactly as they are in the following table. All attribute entries have to be written in a new line with nothing preceding the attributes name.

Comments in the file should be defined with a starting ‘#’ in the line. All NON EMPTY wrongly formatted or non-commented lines will be interpreted as an error and the application will stop its work. Properly formatted lines containing UNKNOWN attributes will also be interpreted as an error.

**Table 3: PIA configuration attributes**

attribute name	Attribute description
----------------	-----------------------

contact_criterion	<p>Criterion by which amino acid contacts should be detected. This attribute can have following integer values:</p> <ul style="list-style-type: none"> <li>• 1 – <i>Maximum Distance</i> criterion</li> <li>• 2 – <i>Van der Waals Distance</i> criterion</li> <li>• 3 – <i>ASA Change</i> criterion</li> <li>• 4 – <i>PIADA</i> criterion</li> </ul> <p>This argument <b>has to</b> be followed by none ore more attributes in a defined order.</p>
CRITERION_ATTRIBUTES	<p>If contact_criterion has values 1 or 2 it has to be followed by “threshold:” argument which defines the criterion distance (e.g. threshold: 4.5).</p> <p>If contact_criterion has value 3 it has to be followed by following attributes <b>respectively</b>: “delta_asa:” defines the minimum ASA change for determining contact residues (e.g. delta_asa: 1.0); “z_slice:” and “r_solvent:” are values used for ASA calculations.</p> <p>If contact_criterion has value 4 no attributes are needed.</p>
radii_filename	Path to radii file to be used.
write_contacts, write_binding_residues, write_binding state	When these arguments have values != 0 output files for contacts, binding_residues and/or binding_state are going to be created.
write_xml, write_table	If set to value != 0 the result will be written in xml and/or table format.
output_dir	Path to the output directory.

## 4.3 File-list file

File-list file contains absolute paths to *Protein Data Bank* files to be used for analysis. Each file path has to be in a separate line. Example of a file-list file for MS Windows version is as follows:

```
C:\pdb_repository\1TIM.pdb
C:\pdb_repository\1AY7.pdb
C:\psaia\pdb_files\1EAW.ent
```

## 5 Examples of output files

### 5.1 PSA Xml output file

```
<?xml version="1.0" encoding="us-ascii"?>
<cPdbPeptide xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns="http://www.fer.hr/lss/namespaces/PDB">
  <pdb_id>1A00</pdb_id>
  <analysis_type>bound</analysis_type>
  <radii_file>C:\psaia\amac_data\chothia.radii</radii_file>
  <ASA_zSice>0.5</ASA_zSice>
  <ASA_rSolvent>1.4</ASA_rSolvent>
  <rasa_filename>C:\psaia\amac_data\natural_asa.asa</rasa_filename>
  <CX_sRadius>10</CX_sRadius>
  <CX_vAtom>20.1</CX_vAtom>
  <hydrophobicity_filename>C:\psaia\amac_data\hydrophobicity.hpb</hydrophobicity_filename>
  <asa>
    <total>37608.2</total>
    <backbone>8612.68</backbone>
    <side_chain>28995.5</side_chain>
    <non_polar>21322.4</non_polar>
    <polar>16285.8</polar>
  </asa>
  <chains>
    <chain>
      <serial_id>A</serial_id>
      <asa>
        <total>5884.32</total>
        <backbone>1283.83</backbone>
        <side_chain>4600.49</side_chain>
        <non_polar>3332.25</non_polar>
        <polar>2552.07</polar>
      </asa>
      <residues>
        <residue>
          <name>ALA</name>
          <serial_id>2</serial_id>
          <asa>
            <total>60.7219</total>
            <backbone>42.1896</backbone>
            <side_chain>18.5323</side_chain>
            <non_polar>21.4489</non_polar>
            <polar>39.273</polar>
          </asa>
        </residue>
      </residues>
    </chain>
  </chains>
</cPdbPeptide>
```

```

    <relative_asa>
      <total>56.6224</total>
      <backbone>97.3906</backbone>
      <side_chain>28.993</side_chain>
      <non_polar>28.2</non_polar>
      <polar>125.996</polar>
    </relative_asa>
    <depth_index>
      <average>0</average>
      <average_deviation>0</average_deviation>
      <average_side_chain>0</average_side_chain>
      <average_side_chain_deviation>0</average_side_chain_deviation>
      <maximum>0</maximum>
      <minimum>0</minimum>
    </depth_index>
    <protrusion_index>
      <average>1.20916</average>
      <average_deviation>0.284475</average_deviation>
      <average_side_chain>0.766081</average_side_chain>
      <average_side_chain_deviation>0</average_side_chain_deviation>
      <maximum>1.60497</maximum>
      <minimum>0.766081</minimum>
    </protrusion_index>
    <hydrophobicity>1.8</hydrophobicity>
  </residue>
  ...
  ...
  ...
</residues>
</chain>
</chains>
</cPdbPeptide>

```

## 5.2 PIA Xml contacts output file

```

<?xml version="1.0" encoding="us-ascii"?>
<peptideResidueContacts xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns="http://www.fer.hr/lss/namespaces/PDB">
  <pdb_id>1A00</pdb_id>
  <radii_filename></radii_filename>
  <contacts>
    <contact>
      <interaction_criterion>PIADA</interaction_criterion>
      <chain_id_1>A</chain_id_1>
      <residue_id_1>90</residue_id_1>
      <residue_name_1>ALA</residue_name_1>
      <chain_id_2>B</chain_id_2>
      <residue_id_2>181</residue_id_2>
      <residue_name_2>HIS</residue_name_2>
      <contact_types>
        <contact_type>vanDerWaals</contact_type>
      </contact_types>
    </contact>
    ...
    ...
    ...
  </contacts>

```

```
</peptideResidueContacts>
```

### 5.3 PIA Xml binding residues output file

```
<?xml version="1.0" encoding="us-ascii"?>
<peptideBindingResidues xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns="http://www.fer.hr/lss/namespaces/PDB">
  <pdb_id>1A00</pdb_id>
  <radii_filename></radii_filename>
  <binding_residues>
    <binding_residue>
      <interaction_criterion>PIADA</interaction_criterion>
      <chain_id>A</chain_id>
      <residue_id>90</residue_id>
      <residue_name>ALA</residue_name>
    </binding_residue>
    ...
    ...
    ...
  </binding_residues>
</peptideBindingResidues>
```

### 5.4 PIA Xml binding status output file

```
<?xml version="1.0" encoding="us-ascii"?>
<peptideResidueBindingStatus xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns="http://www.fer.hr/lss/namespaces/PDB">
  <pdb_id>1A00</pdb_id>
  <radii_filename></radii_filename>
  <residues>
    <residue>
      <interaction_criterion>PIADA</interaction_criterion>
      <chain_id>A</chain_id>
      <residue_id>2</residue_id>
      <residue_name>ALA</residue_name>
      <binding_status>0</binding_status>
    </residue>
    ...
    ...
    ...
  </residues>
</peptideResidueBindingStatus>
```