

Prog2HF

Generated by Doxygen 1.8.15

## 1 Hierarchical Index

### 1.1 Class Hierarchy

This inheritance list is sorted roughly, but not completely, alphabetically:

<b>Build</b>	<b>3</b>
<b>CompatibilityList</b>	<b>5</b>
<b>Inventory</b>	<b>12</b>
<b>Orders</b>	<b>15</b>
<b>Part</b>	<b>16</b>
<b>Case</b>	<b>4</b>
<b>CPU</b>	<b>7</b>
<b>GPU</b>	<b>9</b>
<b>MOBO</b>	<b>14</b>
<b>PSU</b>	<b>19</b>
<b>RAM</b>	<b>20</b>
<b>Storage</b>	<b>23</b>
<b>HDD</b>	<b>10</b>
<b>SSD</b>	<b>22</b>
<b>String</b>	<b>25</b>
<b>TemplInput</b>	<b>30</b>
<b>utos_ostream</b>	<b>33</b>
<b>utos_t</b>	<b>34</b>

## 2 Class Index

### 2.1 Class List

Here are the classes, structs, unions and interfaces with brief descriptions:

<b>Build</b>	<b>3</b>
<b>Case</b>	
<b>Ház</b>	<b>4</b>
<b>CompatibilityList</b>	<b>5</b>
<b>CPU</b>	
<b>Processzor</b>	<b>7</b>

<b>GPU</b>	
Videókártya	9
<b>HDD</b>	
Merevlemez	10
<b>Inventory</b>	12
<b>MOBO</b>	
Alaplap	14
<b>Orders</b>	15
<b>Part</b>	
Alap alkatrész típus	16
<b>PSU</b>	
Táp	19
<b>RAM</b>	
Memória	20
<b>SSD</b>	
SSD (p. 22)	22
<b>Storage</b>	
Tárhely alap	23
<b>String</b>	25
<b>TemplInput</b>	
Lehetséges inputokat tárolja adatokkal való konstruáláshoz	30
<b>utos_ostream</b>	
Szóközösítő stream manipulator	33
<b>utos_t</b>	
Szóközösítő toggle	34

## 3 File Index

### 3.1 File List

Here is a list of all files with brief descriptions:

<b>C:/Users/cxl20/Documents/Visual Studio 2017/Prog2HF/Prog2HF/ atest.cpp</b>	<b>34</b>
<b>C:/Users/cxl20/Documents/Visual Studio 2017/Prog2HF/Prog2HF/ atest.h</b>	<b>35</b>
<b>C:/Users/cxl20/Documents/Visual Studio 2017/Prog2HF/Prog2HF/ Builds.cpp</b>	<b>36</b>
<b>C:/Users/cxl20/Documents/Visual Studio 2017/Prog2HF/Prog2HF/ Builds.h</b>	<b>36</b>
<b>C:/Users/cxl20/Documents/Visual Studio 2017/Prog2HF/Prog2HF/ Compatibility.cpp</b>	<b>37</b>
<b>C:/Users/cxl20/Documents/Visual Studio 2017/Prog2HF/Prog2HF/ Compatibility.h</b>	<b>38</b>
<b>C:/Users/cxl20/Documents/Visual Studio 2017/Prog2HF/Prog2HF/ Inventory.cpp</b>	<b>38</b>

<b>C:/Users/cxl20/Documents/Visual Studio 2017/Prog2HF/Prog2HF/ Inventory.h</b>	<b>??</b>
<b>C:/Users/cxl20/Documents/Visual Studio 2017/Prog2HF/Prog2HF/ main.cpp</b>	<b>??</b>
<b>C:/Users/cxl20/Documents/Visual Studio 2017/Prog2HF/Prog2HF/ main.h</b>	<b>??</b>
<b>C:/Users/cxl20/Documents/Visual Studio 2017/Prog2HF/Prog2HF/ Parts.cpp</b>	<b>??</b>
<b>C:/Users/cxl20/Documents/Visual Studio 2017/Prog2HF/Prog2HF/ Parts.h</b>	<b>??</b>
<b>C:/Users/cxl20/Documents/Visual Studio 2017/Prog2HF/Prog2HF/ schtring.cpp</b>	<b>??</b>
<b>C:/Users/cxl20/Documents/Visual Studio 2017/Prog2HF/Prog2HF/ schtring.hpp</b>	<b>??</b>
<b>C:/Users/cxl20/Documents/Visual Studio 2017/Prog2HF/Prog2HF/ SFML_test.cpp</b>	<b>??</b>

## 4 Class Documentation

### 4.1 Build Class Reference

```
#include <Builds.h>
```

#### Public Member Functions

- **Build** (size\_t capacity=7)
- **~Build** ()
- template<typename T >  
void **push\_back** (T \*part)
- const **Part** \* **operator[]** (int idx) const
- **Part** \* **operator[]** (int idx)

#### 4.1.1 Constructor & Destructor Documentation

##### 4.1.1.1 Build()

```
Build::Build (
    size_t capacity = 7 ) [inline]
```

##### 4.1.1.2 ~Build()

```
Build::~~Build ( ) [inline]
```

#### 4.1.2 Member Function Documentation

#### 4.1.2.1 operator[]() [1/2]

```
const Part* Build::operator[] (
    int idx ) const [inline]
```

#### 4.1.2.2 operator[]() [2/2]

```
Part* Build::operator[] (
    int idx ) [inline]
```

#### 4.1.2.3 push\_back()

```
template<typename T >
void Build::push_back (
    T * part )
```

The documentation for this class was generated from the following files:

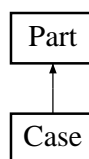
- C:/Users/cxl20/Documents/Visual Studio 2017/Prog2HF/Prog2HF/ **Builds.h**
- C:/Users/cxl20/Documents/Visual Studio 2017/Prog2HF/Prog2HF/ **Builds.cpp**

## 4.2 Case Class Reference

Ház.

```
#include <Parts.h>
```

Inheritance diagram for Case:



#### Public Member Functions

- **Case** ( **String** *brand*, **String** *type*, int *price*, **String** *formfactor*)
- **Case** ( **TemplInput** *tmp*)
- void **print** (std::ostream &*os*) const
- void **print** ( **utos\_ostream** &*tos*) const

#### Additional Inherited Members

#### 4.2.1 Detailed Description

Ház.

## 4.2.2 Constructor & Destructor Documentation

### 4.2.2.1 Case() [1/2]

```
Case::Case (
    String brand,
    String type,
    int price,
    String formfactor ) [inline], [explicit]
```

### 4.2.2.2 Case() [2/2]

```
Case::Case (
    TempInput tmp ) [inline], [explicit]
```

## 4.2.3 Member Function Documentation

### 4.2.3.1 print() [1/2]

```
void Case::print (
    std::ostream & os ) const [virtual]
```

Reimplemented from **Part** (p. 18).

### 4.2.3.2 print() [2/2]

```
void Case::print (
    utos_ostream & tos ) const [virtual]
```

Reimplemented from **Part** (p. 18).

The documentation for this class was generated from the following files:

- C:/Users/cxl20/Documents/Visual Studio 2017/Prog2HF/Prog2HF/ **Parts.h**
- C:/Users/cxl20/Documents/Visual Studio 2017/Prog2HF/Prog2HF/ **Parts.cpp**

## 4.3 CompatibilityList Class Reference

```
#include <Compatibility.h>
```

## Public Member Functions

- **CompatibilityList** ()
- **CompatibilityList** ( **String** &)
- **~CompatibilityList** ()
- int **get\_length** () const
- **String** \* **get\_listp** () const
- void **addItems** ( **String** &)
- bool **operator==** ( **String** &rhs)
- bool **operator==** (const char \*rhs)

### 4.3.1 Constructor & Destructor Documentation

#### 4.3.1.1 CompatibilityList() [1/2]

```
CompatibilityList::CompatibilityList ( ) [inline], [explicit]
```

#### 4.3.1.2 CompatibilityList() [2/2]

```
CompatibilityList::CompatibilityList (
    String & slist ) [explicit]
```

#### 4.3.1.3 ~CompatibilityList()

```
CompatibilityList::~~CompatibilityList ( ) [inline]
```

### 4.3.2 Member Function Documentation

#### 4.3.2.1 addItems()

```
void CompatibilityList::addItems (
    String & slist )
```

#### 4.3.2.2 get\_length()

```
int CompatibilityList::get_length ( ) const [inline]
```

## 4.3.2.3 get\_listp()

```
String* CompatibilityList::get_listp ( ) const [inline]
```

## 4.3.2.4 operator==( ) [1/2]

```
bool CompatibilityList::operator== (
    String & rhs ) [inline]
```

## 4.3.2.5 operator==( ) [2/2]

```
bool CompatibilityList::operator== (
    const char * rhs ) [inline]
```

The documentation for this class was generated from the following files:

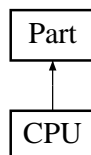
- C:/Users/cxl20/Documents/Visual Studio 2017/Prog2HF/Prog2HF/ **Compatibility.h**
- C:/Users/cxl20/Documents/Visual Studio 2017/Prog2HF/Prog2HF/ **Compatibility.cpp**

## 4.4 CPU Class Reference

Processzor.

```
#include <Parts.h>
```

Inheritance diagram for CPU:



## Public Member Functions

- **CPU** ( **String** brand, **String** type, int price, int clk, int cores, **String** socket, bool multithreading)
- **CPU** ( **TemplInput** &tmp)
- void **print** (std::ostream &os) const
- void **print** ( **utos\_ostream** &tos) const
- void **testh** () const

## Additional Inherited Members

## 4.4.1 Detailed Description

Processzor.



## 4.4.2 Constructor & Destructor Documentation

### 4.4.2.1 CPU() [1/2]

```
CPU::CPU (
    String brand,
    String type,
    int price,
    int clk,
    int cores,
    String socket,
    bool multithreading ) [inline], [explicit]
```

### 4.4.2.2 CPU() [2/2]

```
CPU::CPU (
    TempInput & tmp ) [inline], [explicit]
```

## 4.4.3 Member Function Documentation

### 4.4.3.1 print() [1/2]

```
void CPU::print (
    std::ostream & os ) const [virtual]
```

Reimplemented from **Part** (p. 18).

### 4.4.3.2 print() [2/2]

```
void CPU::print (
    utos_ostream & tos ) const [virtual]
```

Reimplemented from **Part** (p. 18).

### 4.4.3.3 testh()

```
void CPU::testh ( ) const
```

The documentation for this class was generated from the following files:

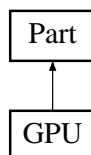
- C:/Users/cxl20/Documents/Visual Studio 2017/Prog2HF/Prog2HF/**Parts.h**
- C:/Users/cxl20/Documents/Visual Studio 2017/Prog2HF/Prog2HF/**Parts.cpp**

## 4.5 GPU Class Reference

Videókártya.

```
#include <Parts.h>
```

Inheritance diagram for GPU:



### Public Member Functions

- **GPU** ( **String** *brand*, **String** *type*, int *price*, int *clk*, int *vram*)
- **GPU** ( **TemplInput** &tmp)
- void **print** (std::ostream &os) const
- void **print** ( **utos\_ostream** &tos) const

### Additional Inherited Members

#### 4.5.1 Detailed Description

Videókártya.

#### 4.5.2 Constructor & Destructor Documentation

##### 4.5.2.1 GPU() [1/2]

```
GPU::GPU (
    String brand,
    String type,
    int price,
    int clk,
    int vram ) [inline], [explicit]
```

##### 4.5.2.2 GPU() [2/2]

```
GPU::GPU (
    TemplInput & tmp ) [inline], [explicit]
```

### 4.5.3 Member Function Documentation

#### 4.5.3.1 `print()` [1/2]

```
void GPU::print (
    std::ostream & os ) const [virtual]
```

Reimplemented from **Part** (p. 18).

#### 4.5.3.2 `print()` [2/2]

```
void GPU::print (
    utos_ostream & tos ) const [virtual]
```

Reimplemented from **Part** (p. 18).

The documentation for this class was generated from the following files:

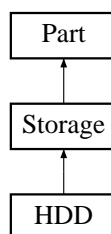
- C:/Users/cxl20/Documents/Visual Studio 2017/Prog2HF/Prog2HF/ **Parts.h**
- C:/Users/cxl20/Documents/Visual Studio 2017/Prog2HF/Prog2HF/ **Parts.cpp**

## 4.6 HDD Class Reference

Merevlemez.

```
#include <Parts.h>
```

Inheritance diagram for HDD:



#### Public Member Functions

- **HDD** ( **String** brand, **String** type, int price, int size, int readspeed, int writespeed, int rpm)
- **HDD** ( **TemplInput** tmp)
- void **print** (std::ostream &os) const
- void **print** ( **utos\_ostream** &tos) const

## Additional Inherited Members

### 4.6.1 Detailed Description

Merevlemez.

### 4.6.2 Constructor & Destructor Documentation

#### 4.6.2.1 HDD() [1/2]

```
HDD::HDD (
    String brand,
    String type,
    int price,
    int size,
    int readspeed,
    int writespeed,
    int rpm ) [inline], [explicit]
```

#### 4.6.2.2 HDD() [2/2]

```
HDD::HDD (
    TempInput tmp ) [inline], [explicit]
```

### 4.6.3 Member Function Documentation

#### 4.6.3.1 print() [1/2]

```
void HDD::print (
    std::ostream & os ) const [virtual]
```

Reimplemented from **Storage** (p. 24).

#### 4.6.3.2 print() [2/2]

```
void HDD::print (
    utos_ostream & tos ) const [virtual]
```

Reimplemented from **Storage** (p. 24).

The documentation for this class was generated from the following files:

- C:/Users/cxl20/Documents/Visual Studio 2017/Prog2HF/Prog2HF/ **Parts.h**
- C:/Users/cxl20/Documents/Visual Studio 2017/Prog2HF/Prog2HF/ **Parts.cpp**

## 4.7 Inventory Class Reference

```
#include <Inventory.h>
```

### Public Member Functions

- **Inventory** (size\_t capacity=1)
- **~Inventory** ()
- int **get\_size** ()
- void **loadPart** (std::istream &is, **TempInput** &tmp, **enumPart**)
- void **saveInventory** (std::ostream &os)
- void **printInventory** (std::ostream &os)
- void **removePart** (int a)
- template<typename T >  
void **push\_back** (T \*part)
- const **Part** \* **operator[]** (int idx) const
- **Part** \* **operator[]** (int idx)

### 4.7.1 Constructor & Destructor Documentation

#### 4.7.1.1 Inventory()

```
Inventory::Inventory (  
    size_t capacity = 1 ) [inline]
```

#### 4.7.1.2 ~Inventory()

```
Inventory::~Inventory ( ) [inline]
```

### 4.7.2 Member Function Documentation

#### 4.7.2.1 get\_size()

```
int Inventory::get_size ( ) [inline]
```

#### 4.7.2.2 loadPart()

```
void Inventory::loadPart (  
    std::istream & is,  
    TempInput & tmp,  
    enumPart e )
```

#### 4.7.2.3 operator[]() [1/2]

```
const Part* Inventory::operator[] (
    int idx ) const [inline]
```

#### 4.7.2.4 operator[]() [2/2]

```
Part* Inventory::operator[] (
    int idx ) [inline]
```

#### 4.7.2.5 printInventory()

```
void Inventory::printInventory (
    std::ostream & os )
```

#### 4.7.2.6 push\_back()

```
template<typename T >
void Inventory::push_back (
    T * part )
```

#### 4.7.2.7 removePart()

```
void Inventory::removePart (
    int a )
```

#### 4.7.2.8 saveInventory()

```
void Inventory::saveInventory (
    std::ostream & os )
```

The documentation for this class was generated from the following files:

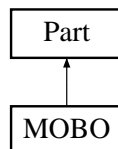
- C:/Users/cxl20/Documents/Visual Studio 2017/Prog2HF/Prog2HF/ **Inventory.h**
- C:/Users/cxl20/Documents/Visual Studio 2017/Prog2HF/Prog2HF/ **Inventory.cpp**

## 4.8 MOBO Class Reference

Alaplap.

```
#include <Parts.h>
```

Inheritance diagram for MOBO:



### Public Member Functions

- **MOBO** ( **String** brand, **String** type, int price, **String** socket, **String** chipset, **String** formfactor)
- **MOBO** ( **TemplInput** tmp)
- void **print** (std::ostream &os) const
- void **print** ( **utos\_ostream** &tos) const

### Additional Inherited Members

#### 4.8.1 Detailed Description

Alaplap.

#### 4.8.2 Constructor & Destructor Documentation

##### 4.8.2.1 MOBO() [1/2]

```
MOBO::MOBO (
    String brand,
    String type,
    int price,
    String socket,
    String chipset,
    String formfactor ) [inline], [explicit]
```

##### 4.8.2.2 MOBO() [2/2]

```
MOBO::MOBO (
    TemplInput tmp ) [inline], [explicit]
```

## 4.8.3 Member Function Documentation

## 4.8.3.1 print() [1/2]

```
void MOBO::print (
    std::ostream & os ) const [virtual]
```

Reimplemented from **Part** (p. 18).

## 4.8.3.2 print() [2/2]

```
void MOBO::print (
    utos_ostream & tos ) const [virtual]
```

Reimplemented from **Part** (p. 18).

The documentation for this class was generated from the following files:

- C:/Users/cxl20/Documents/Visual Studio 2017/Prog2HF/Prog2HF/ **Parts.h**
- C:/Users/cxl20/Documents/Visual Studio 2017/Prog2HF/Prog2HF/ **Parts.cpp**

## 4.9 Orders Class Reference

```
#include <Builds.h>
```

## Public Member Functions

- **Orders** (size\_t capacity=1)
- **~Orders** ()
- template<typename T >  
void **push\_back** (T \*part)
- const **Build operator[]** (int idx) const
- **Build operator[]** (int idx)

## 4.9.1 Constructor &amp; Destructor Documentation

## 4.9.1.1 Orders()

```
Orders::Orders (
    size_t capacity = 1 ) [inline]
```



#### 4.9.1.2 ~Orders()

```
Orders::~~Orders ( ) [inline]
```

### 4.9.2 Member Function Documentation

#### 4.9.2.1 operator[]() [1/2]

```
const Build Orders::operator[] (
    int idx ) const [inline]
```

#### 4.9.2.2 operator[]() [2/2]

```
Build Orders::operator[] (
    int idx ) [inline]
```

#### 4.9.2.3 push\_back()

```
template<typename T >
void Orders::push_back (
    T * part )
```

The documentation for this class was generated from the following files:

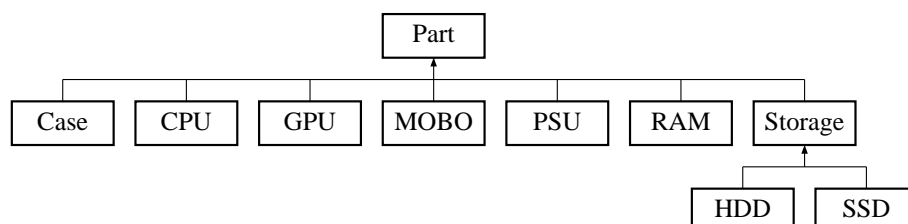
- C:/Users/cxl20/Documents/Visual Studio 2017/Prog2HF/Prog2HF/ **Builds.h**
- C:/Users/cxl20/Documents/Visual Studio 2017/Prog2HF/Prog2HF/ **Builds.cpp**

## 4.10 Part Class Reference

Alap alkatrész típus.

```
#include <Parts.h>
```

Inheritance diagram for Part:



## Public Member Functions

- **Part** ( **String** brand="", **String** type="", int price=0)
- virtual **~Part** ()
- **String** **get\_brand** ()
- **String** **get\_type** ()
- int **get\_price** ()
- virtual void **print** (std::ostream &os) const
- virtual void **print** ( **utos\_ostream** &tos) const

## Protected Attributes

- **String** **brand**  
*Gyártó*
- **String** **type**  
*Típus.*
- int **price**  
*Ár.*

## 4.10.1 Detailed Description

Alap alkatrész típus.

## 4.10.2 Constructor &amp; Destructor Documentation

## 4.10.2.1 Part()

```
Part::Part (
    String brand = "",
    String type = "",
    int price = 0 ) [inline]
```

## 4.10.2.2 ~Part()

```
virtual Part::~~Part ( ) [inline], [virtual]
```

## 4.10.3 Member Function Documentation

## 4.10.3.1 get\_brand()

```
String Part::get_brand ( ) [inline]
```

#### 4.10.3.2 `get_price()`

```
int Part::get_price ( ) [inline]
```

#### 4.10.3.3 `get_type()`

```
String Part::get_type ( ) [inline]
```

#### 4.10.3.4 `print()` [1/2]

```
void Part::print (
    std::ostream & os ) const [virtual]
```

Reimplemented in **HDD** (p. 11), **SSD** (p. 23), **Storage** (p. 24), **PSU** (p. 20), **Case** (p. 5), **RAM** (p. 21), **MOBO** (p. 15), **GPU** (p. 10), and **CPU** (p. 8).

#### 4.10.3.5 `print()` [2/2]

```
void Part::print (
    utos_ostream & tos ) const [virtual]
```

Reimplemented in **HDD** (p. 11), **SSD** (p. 23), **Storage** (p. 24), **PSU** (p. 20), **Case** (p. 5), **RAM** (p. 21), **MOBO** (p. 15), **GPU** (p. 10), and **CPU** (p. 8).

### 4.10.4 Member Data Documentation

#### 4.10.4.1 `brand`

```
String Part::brand [protected]
```

Gyártó

#### 4.10.4.2 `price`

```
int Part::price [protected]
```

Ár.

## 4.10.4.3 type

```
String Part::type [protected]
```

Típus.

The documentation for this class was generated from the following files:

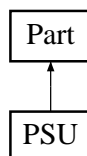
- C:/Users/cxl20/Documents/Visual Studio 2017/Prog2HF/Prog2HF/ **Parts.h**
- C:/Users/cxl20/Documents/Visual Studio 2017/Prog2HF/Prog2HF/ **Parts.cpp**

## 4.11 PSU Class Reference

Táp.

```
#include <Parts.h>
```

Inheritance diagram for PSU:



## Public Member Functions

- **PSU** ( **String** brand, **String** type, int price, int wattage)
- **PSU** ( **TemplInput** tmp)
- void **print** (std::ostream &os) const
- void **print** ( **utos\_ostream** &tos) const

## Additional Inherited Members

## 4.11.1 Detailed Description

Táp.

## 4.11.2 Constructor &amp; Destructor Documentation

## 4.11.2.1 PSU() [1/2]

```
PSU::PSU (
    String brand,
    String type,
    int price,
    int wattage ) [inline], [explicit]
```

#### 4.11.2.2 PSU() [2/2]

```
PSU::PSU (
    TempInput tmp ) [inline], [explicit]
```

### 4.11.3 Member Function Documentation

#### 4.11.3.1 print() [1/2]

```
void PSU::print (
    std::ostream & os ) const [virtual]
```

Reimplemented from **Part** (p. 18).

#### 4.11.3.2 print() [2/2]

```
void PSU::print (
    utos_ostream & tos ) const [virtual]
```

Reimplemented from **Part** (p. 18).

The documentation for this class was generated from the following files:

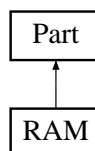
- C:/Users/cxl20/Documents/Visual Studio 2017/Prog2HF/Prog2HF/ **Parts.h**
- C:/Users/cxl20/Documents/Visual Studio 2017/Prog2HF/Prog2HF/ **Parts.cpp**

## 4.12 RAM Class Reference

Memória.

```
#include <Parts.h>
```

Inheritance diagram for RAM:



#### Public Member Functions

- **RAM** ( **String** brand, **String** type, int price, int clk, int size)
- **RAM** ( **TempInput** tmp)
- void **print** (std::ostream &os) const
- void **print** ( **utos\_ostream** &tos) const

## Additional Inherited Members

### 4.12.1 Detailed Description

Memória.

### 4.12.2 Constructor & Destructor Documentation

#### 4.12.2.1 RAM() [1/2]

```
RAM::RAM (
    String brand,
    String type,
    int price,
    int clk,
    int size ) [inline], [explicit]
```

#### 4.12.2.2 RAM() [2/2]

```
RAM::RAM (
    TempInput tmp ) [inline], [explicit]
```

### 4.12.3 Member Function Documentation

#### 4.12.3.1 print() [1/2]

```
void RAM::print (
    std::ostream & os ) const [virtual]
```

Reimplemented from **Part** (p. 18).

#### 4.12.3.2 print() [2/2]

```
void RAM::print (
    utos_ostream & tos ) const [virtual]
```

Reimplemented from **Part** (p. 18).

The documentation for this class was generated from the following files:

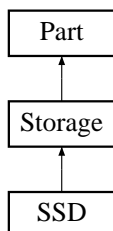
- C:/Users/cxl20/Documents/Visual Studio 2017/Prog2HF/Prog2HF/**Parts.h**
- C:/Users/cxl20/Documents/Visual Studio 2017/Prog2HF/Prog2HF/**Parts.cpp**

### 4.13 SSD Class Reference

**SSD** (p. 22).

```
#include <Parts.h>
```

Inheritance diagram for SSD:



#### Public Member Functions

- **SSD** ( **String** brand, **String** type, int price, int size, int readspeed, int writespeed, **String** formfactor, **String** flashtype)
- **SSD** ( **TemplInput** tmp)
- void **print** (std::ostream &os) const
- void **print** ( **utos\_ostream** &tos) const

#### Additional Inherited Members

##### 4.13.1 Detailed Description

**SSD** (p. 22).

##### 4.13.2 Constructor & Destructor Documentation

###### 4.13.2.1 SSD() [1/2]

```
SSD::SSD (
    String brand,
    String type,
    int price,
    int size,
    int readspeed,
    int writespeed,
    String formfactor,
    String flashtype ) [inline], [explicit]
```

## 4.13.2.2 SSD() [2/2]

```
SSD::SSD (
    TempInput tmp ) [inline], [explicit]
```

## 4.13.3 Member Function Documentation

## 4.13.3.1 print() [1/2]

```
void SSD::print (
    std::ostream & os ) const [virtual]
```

Reimplemented from **Storage** (p. 24).

## 4.13.3.2 print() [2/2]

```
void SSD::print (
    utos_ostream & tos ) const [virtual]
```

Reimplemented from **Storage** (p. 24).

The documentation for this class was generated from the following files:

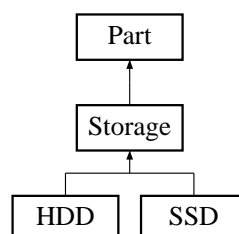
- C:/Users/cxl20/Documents/Visual Studio 2017/Prog2HF/Prog2HF/ **Parts.h**
- C:/Users/cxl20/Documents/Visual Studio 2017/Prog2HF/Prog2HF/ **Parts.cpp**

## 4.14 Storage Class Reference

Tárhely alap.

```
#include <Parts.h>
```

Inheritance diagram for Storage:



## Public Member Functions

- **Storage** ( **String** brand, **String** type, int price, int size, int readspeed, int writespeed)
- virtual void **print** (std::ostream &os) const
- virtual void **print** ( utos\_ostream &tos) const



## Protected Attributes

- int **size**  
*Méret.*
- int **readspeed**  
*Olvasási sebesség.*
- int **writespeed**  
*Írási sebesség.*

### 4.14.1 Detailed Description

Tárhely alap.

### 4.14.2 Constructor & Destructor Documentation

#### 4.14.2.1 Storage()

```
Storage::Storage (
    String brand,
    String type,
    int price,
    int size,
    int readspeed,
    int writespeed ) [inline], [explicit]
```

### 4.14.3 Member Function Documentation

#### 4.14.3.1 print() [1/2]

```
void Storage::print (
    std::ostream & os ) const [virtual]
```

Reimplemented from **Part** (p. 18).

Reimplemented in **HDD** (p. 11), and **SSD** (p. 23).

#### 4.14.3.2 print() [2/2]

```
void Storage::print (
    utos_ostream & tos ) const [virtual]
```

Reimplemented from **Part** (p. 18).

Reimplemented in **HDD** (p. 11), and **SSD** (p. 23).

#### 4.14.4 Member Data Documentation

##### 4.14.4.1 readspeed

```
int Storage::readspeed [protected]
```

Olvasási sebesség.

##### 4.14.4.2 size

```
int Storage::size [protected]
```

Méret.

##### 4.14.4.3 writespeed

```
int Storage::writespeed [protected]
```

Írási sebesség.

The documentation for this class was generated from the following files:

- C:/Users/cxl20/Documents/Visual Studio 2017/Prog2HF/Prog2HF/**Parts.h**
- C:/Users/cxl20/Documents/Visual Studio 2017/Prog2HF/Prog2HF/**Parts.cpp**

## 4.15 String Class Reference

```
#include <schtring.hpp>
```

### Public Member Functions

- **size\_t size ()** const  
*hossz lezáró nulla nélkül*
- **size\_t length ()** const  
*Visszaadja a string hosszát.*
- **String ()**  
*Default konstruktor.*
- **String (char ch)**  
*Konstruktor: egy char karakterre.*
- **String (const char \*p)**  
*Konstruktor: egy karakter tömbre.*
- **String (const String &s1)**  
*Konstruktor: egy másik Stringre.*
- **const char \* c\_str ()** const

- C-stringet ad vissza.*
- **~String ()**  
*Destruktor.*
- **String & operator=** (const **String** &rhs\_s)  
*Egyenlőség operator.*
- **String & operator+=** (const **String** &rhs\_s)  
*Pluszegyenlő operator.*
- **String operator+** (const **String** &rhs\_s) const  
*string + string*
- **String operator+** (char rhs\_c)  
*string + karakter*
- **bool operator==** ( **String** &rhs\_s)  
*hasonlító operator stringgel*
- **bool operator==** (const char \*rhs\_s)  
*hasonlító operator char tömbbel*
- **bool operator==** (const char \*rhs\_s) const
- **String operator--** (int a)  
*kitörli az utolsó karaktert a stringből*
- **char & operator[]** (unsigned int idx)  
*index operator*
- **const char & operator[]** (unsigned int idx) const  
*index operator*
- **void erase** ()  
*törli a stringben lévő karaktereket*
- **void removeFirstX** (int x)  
*törli az első x karaktert a stringből*

#### 4.15.1 Constructor & Destructor Documentation

##### 4.15.1.1 String() [1/4]

```
String::String ( ) [inline]
```

Default konstruktor.

##### 4.15.1.2 String() [2/4]

```
String::String (
    char ch )
```

Konstruktor: egy char karakterre.

#### 4.15.1.3 String() [3/4]

```
String::String (
    const char * p )
```

Konstruktor: egy karakter tömbre.

#### 4.15.1.4 String() [4/4]

```
String::String (
    const String & s1 )
```

Konstruktor: egy másik Stringre.

#### 4.15.1.5 ~String()

```
String::~~String ( ) [inline]
```

Destruktor.

### 4.15.2 Member Function Documentation

#### 4.15.2.1 c\_str()

```
const char* String::c_str ( ) const [inline]
```

C-stringet ad vissza.

#### 4.15.2.2 erase()

```
void String::erase ( ) [inline]
```

törli a stringben lévő karaktereket

#### 4.15.2.3 length()

```
size_t String::length ( ) const [inline]
```

Visszaadja a string hosszát.

#### 4.15.2.4 operator+() [1/2]

```
String String::operator+ (
    const String & rhs_s ) const
```

string + string

#### 4.15.2.5 operator+() [2/2]

```
String String::operator+ (
    char rhs_c ) [inline]
```

string + karakter

#### 4.15.2.6 operator+=( )

```
String& String::operator+= (
    const String & rhs_s ) [inline]
```

Pluszegyenlő operator.

#### 4.15.2.7 operator--()

```
String String::operator-- (
    int a )
```

kitörli az utolsó karaktert a stringből

#### 4.15.2.8 operator=( )

```
String & String::operator= (
    const String & rhs_s )
```

Egyenlőség operator.

#### 4.15.2.9 operator==( ) [1/3]

```
bool String::operator== (
    String & rhs_s )
```

használati operator stringgel

**4.15.2.10 operator==(** [2/3]

```
bool String::operator==(
    const char * rhs_s )
```

hasonlító operator char tömbbel

**4.15.2.11 operator==(** [3/3]

```
bool String::operator==(
    const char * rhs_s ) const
```

**4.15.2.12 operator[](** [1/2]

```
char & String::operator[] (
    unsigned int idx )
```

index operator

**4.15.2.13 operator[](** [2/2]

```
const char & String::operator[] (
    unsigned int idx ) const
```

index operator

**4.15.2.14 removeFirstX()**

```
void String::removeFirstX (
    int x )
```

törli az első x karaktert a stringből

**4.15.2.15 size()**

```
size_t String::size ( ) const [inline]
```

hossz lezáró nulla nélkül

Visszaadja a string hosszát

The documentation for this class was generated from the following files:

- C:/Users/cxl20/Documents/Visual Studio 2017/Prog2HF/Prog2HF/ **schtring.hpp**
- C:/Users/cxl20/Documents/Visual Studio 2017/Prog2HF/Prog2HF/ **schtring.cpp**

## 4.16 TemplInput Struct Reference

Lehetséges inputokat tárolja adatokkal való konstruáláshoz.

```
#include <Parts.h>
```

### Public Attributes

- **String instruction**  
*Mihez tartozik a változó*
- **String cname**  
*Kompatibilitás lista neve.*
- **String brand**  
*Gyártó*
- **String type**  
*Típus.*
- int **price**  
*Ár.*
- **String socket**  
*Foglalat.*
- int **clk**  
*Órajel.*
- int **cores**  
*Magok száma.*
- bool **multithreading**  
*Multithreading support.*
- **String chipset**  
*Chipset.*
- **String formfactor**  
*Méret szabvány.*
- int **size**  
*Memória méret.*
- int **wattage**  
*Teljesítmény.*
- int **readspeed**  
*Olvasási sebesség.*
- int **writespeed**  
*Írási sebesség.*
- **String flashtype**  
*Flash csip típusa.*
- int **rpm**  
*Fordulatszám.*

### 4.16.1 Detailed Description

Lehetséges inputokat tárolja adatokkal való konstruáláshoz.

#### 4.16.2 Member Data Documentation

##### 4.16.2.1 brand

**String** TemplInput::brand

Gyártó

##### 4.16.2.2 chipset

**String** TemplInput::chipset

Chipset.

##### 4.16.2.3 clk

int TemplInput::clk

Órajel.

##### 4.16.2.4 cname

**String** TemplInput::cname

Kompatibilitás lista neve.

##### 4.16.2.5 cores

int TemplInput::cores

Magok száma.

##### 4.16.2.6 flashtype

**String** TemplInput::flashtype

Flash csip típusa.



#### 4.16.2.7 formfactor

**String** TempInput::formfactor

Méret szabvány.

#### 4.16.2.8 instruction

**String** TempInput::instruction

Mihez tartozik a változó

#### 4.16.2.9 multithreading

**bool** TempInput::multithreading

Multithreading support.

#### 4.16.2.10 price

**int** TempInput::price

Ár.

#### 4.16.2.11 readspeed

**int** TempInput::readspeed

Olvasási sebesség.

#### 4.16.2.12 rpm

**int** TempInput::rpm

Fordulatszám.

#### 4.16.2.13 size

**int** TempInput::size

Memória méret.

## 4.16.2.14 socket

```
String TempInput::socket
```

Foglalat.

## 4.16.2.15 type

```
String TempInput::type
```

Típus.

## 4.16.2.16 wattage

```
int TempInput::wattage
```

Teljesítmény.

## 4.16.2.17 writespeed

```
int TempInput::writespeed
```

Írási sebesség.

The documentation for this struct was generated from the following file:

- C:/Users/cxl20/Documents/Visual Studio 2017/Prog2HF/Prog2HF/**Parts.h**

## 4.17 utos\_ostream Struct Reference

szóközösítő stream manipulator

```
#include <schtring.hpp>
```

## Public Attributes

- std::ostream & **os**

## 4.17.1 Detailed Description

szóközösítő stream manipulator

#### 4.17.2 Member Data Documentation

##### 4.17.2.1 os

```
std::ostream& utos_ostream::os
```

The documentation for this struct was generated from the following file:

- C:/Users/cxl20/Documents/Visual Studio 2017/Prog2HF/Prog2HF/ **schtring.hpp**

#### 4.18 utos\_t Struct Reference

szóközösítő toggle

```
#include <schtring.hpp>
```

##### 4.18.1 Detailed Description

szóközösítő toggle

The documentation for this struct was generated from the following file:

- C:/Users/cxl20/Documents/Visual Studio 2017/Prog2HF/Prog2HF/ **schtring.hpp**

## 5 File Documentation

### 5.1 C:/Users/cxl20/Documents/Visual Studio 2017/Prog2HF/Prog2HF/atest.cpp File Reference

```
#include "atest.h"
```

#### Functions

- void **test1** (std::fstream &partsFile, const char filename[52])  
*Test the if the parts file could be opened.*
- bool **test2** ( **Inventory** &inventory)  
*Test whether the heterogenous collection contains the derived classes.*
- bool **test3** ( **String** test1, **String** test2)  
*Test the non case sensitive **String** (p. 25) compare.*
- bool **test4** ( **String** asd, const char \*test)  
*Test the string shortener.*

### 5.1.1 Function Documentation

#### 5.1.1.1 test1()

```
void test1 (
    std::fstream & partsFile,
    const char filename[52] )
```

Test the if the parts file could be opened.

#### 5.1.1.2 test2()

```
bool test2 (
    Inventory & inventory )
```

Test whether the heterogenous collection contains the derived classes.

#### 5.1.1.3 test3()

```
bool test3 (
    String test1,
    String test2 )
```

Test the non case sensitive **String** (p. 25) compare.

#### 5.1.1.4 test4()

```
bool test4 (
    String asd,
    const char * test )
```

Test the string shortener.

## 5.2 C:/Users/cxl20/Documents/Visual Studio 2017/Prog2HF/Prog2HF/atest.h File Reference

```
#include "main.h"
```

### Functions

- void **test1** (std::fstream &partsFile, const char partsfilename[52])  
*Test the if the parts file could be opened.*
- bool **test2** ( **Inventory** &inventory)  
*Test whether the heterogenous collection contains the derived classes.*
- bool **test3** ( **String** test1, **String** test2)  
*Test the non case sensitive **String** (p. 25) compare.*
- bool **test4** ( **String** asd, const char \*test)  
*Test the string shortener.*

## 5.2.1 Function Documentation

### 5.2.1.1 test1()

```
void test1 (
    std::fstream & partsFile,
    const char partsfilename[52] )
```

Test the if the parts file could be opened.

### 5.2.1.2 test2()

```
bool test2 (
    Inventory & inventory )
```

Test whether the heterogenous collection contains the derived classes.

### 5.2.1.3 test3()

```
bool test3 (
    String test1,
    String test2 )
```

Test the non case sensitive **String** (p. 25) compare.

### 5.2.1.4 test4()

```
bool test4 (
    String asd,
    const char * test )
```

Test the string shortener.

## 5.3 C:/Users/cxl20/Documents/Visual Studio 2017/Prog2HF/Prog2HF/Builds.cpp File Reference

```
#include "Builds.h"
```

## 5.4 C:/Users/cxl20/Documents/Visual Studio 2017/Prog2HF/Prog2HF/Builds.h File Reference

```
#include "Parts.h"
```

## Classes

- class **Build**
- class **Orders**

## Functions

- `std::ostream & operator<< (std::ostream &os, const Build &b)`
- `std::ostream & operator<< (std::ostream &os, const Orders &o)`

## 5.4.1 Function Documentation

5.4.1.1 `operator<<()` [1/2]

```
std::ostream& operator<< (
    std::ostream & os,
    const Build & b )
```

5.4.1.2 `operator<<()` [2/2]

```
std::ostream& operator<< (
    std::ostream & os,
    const Orders & o )
```

## 5.5 C:/Users/cxl20/Documents/Visual Studio 2017/Prog2HF/Prog2HF/Compatibility.cpp File Reference

```
#include "Compatibility.h"
```

## Functions

- `std::ostream & operator<< (std::ostream &os, const CompatibilityList &cl)`

## 5.5.1 Function Documentation

5.5.1.1 `operator<<()`

```
std::ostream& operator<< (
    std::ostream & os,
    const CompatibilityList & cl )
```

## 5.6 C:/Users/cxl20/Documents/Visual Studio 2017/Prog2HF/Prog2HF/Compatibility.h File Reference

```
#include <iostream>
#include "schtring.hpp"
```

### Classes

- class **CompatibilityList**

### Functions

- `std::ostream & operator<< (std::ostream &os, const CompatibilityList &cl)`
- `template<typename T1 = String, typename T2 = String>`  
`bool compatible (T1 is, T2 with, CompatibilityList cl)`

#### 5.6.1 Function Documentation

##### 5.6.1.1 compatible()

```
template<typename T1 = String, typename T2 = String>
bool compatible (
    T1 is,
    T2 with,
    CompatibilityList cl )
```

##### 5.6.1.2 operator<<()

```
std::ostream& operator<< (
    std::ostream & os,
    const CompatibilityList & cl )
```

## 5.7 C:/Users/cxl20/Documents/Visual Studio 2017/Prog2HF/Prog2HF/Inventory.cpp File Reference

```
#include "Inventory.h"
```

### Functions

- void **setEnum** ( **String** inst, enum **enumPart** &e)
- void **LoadParams** (std::istream &is, **TemplInput** &tmp, int const params)