



СРЕДИ ЗА ПРОГРАМИРАНЕ

ПРОГРАМНО ОСИГУРЯВАНЕ НА КОМПЮТЪРА

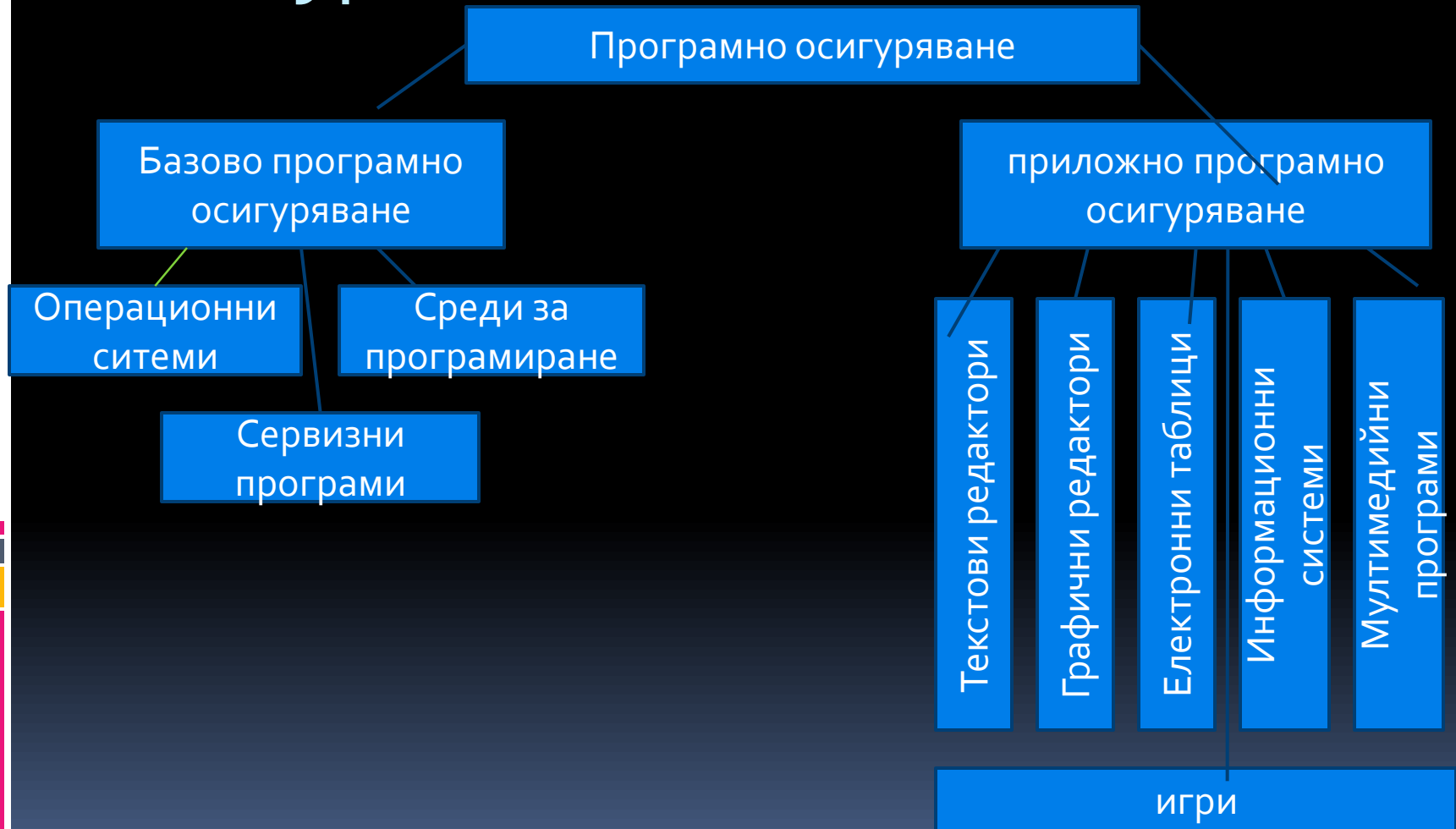
Същност на програмното осигуряване

За да могат компютрите да разбират описаните на езика за програмиране алгоритми, те трябва да бъдат преведени на техния език.

Подадените команди се изпълняват от централния процесор. Той се обръща към оперативната памет, където са записани инструкциите и данните, прочита от там поредната инструкция и необходимите за нея данни и я изпълнява. След това избира нова инструкция, изпълнява я и т.н.

1. Казваме, че съвкупността от всички програми, които работят на даден компютър, образуват неговото програмно осигуряване –софтуер.
2. За апаратната част на компютъра се използва понятието – хардуер.
3. Последователността от команди, които компютъра изпълнява в определен ред се нарича **програма**.

Структура на програмното осигуряване





Елементи на средите за програмиране

СРЕДИ ЗА ПРОГРАМИРАНЕ

Що е програмиране?



-Как компютрите записват въведените данни?

-Какво представлява една програма написана на език за програмиране?

-транслатори – програми, които превеждат изходните програми на езика на компютъра

Елементи на средите за програмиране

```
#include <iostream.h>
#include "welcome.h"
#define my_int signed int

namespace user{
    enum DIGITS {ONE, TWO};

    class Node {
    private:
        Node* toNext;
        char* text;

    public:
        Node(char* s) : text(s), toNext(0){};
        Node* add(char*);
        Node* next();
        char* content();
    };

    Node* Node::next(){
        return toNext;
    }
    char* Node::content() {
        return text;
    }
    Node* Node::add(char* s){
        toNext = new Node(s);
        return toNext;
    }
}

int main(int argc, char**argv) {
    using namespace user;
```

Текстов редактор-програма за:

- ✓ въвеждане на изходната програма
- ✓ редактиране на текста
- ✓ съхраняване и търсене на съхранени вече програми

Транслатор - програма за:

- ✓ Проверява за грешки
- ✓ Превежда изходната програма на езика на компютъра

Интерпретатор-

- ✓ Обработва всяка команда отделно, проверява я за грешки и я изпълнява. Премахва на следващата команда

Компилатор-

- ✓ Обработва цялата програма, проверява я за грешки, превежда я и така получава изпълнима програма

Дебъгери (debuger)

- ✓ Проверява за логически грешки
- ✓ Може да

Библиотеки от стандартни програми

- В структурата на по-сложните обекти присъстват по-прости
- Подчинявайки се на този принцип, още от самото начало може да поставим основите на модулното програмиране.
- Много скоро, след създаване на първите компютърни програми са могли да се вграждат предварително създадени и проверени модули – подпрограми, наричани още стандартни подпрограми(функции)
- Този начин на организация на програмирането има известни предимства. Ако една програма е готова, я използваме без да я пишем отново и да я проверяваме.
- В езиките C и C++ всички функции за извеждане и въвеждане са събрани в библиотеките ***stdio***, ***iostream*** и др, а математическите функции са събрани в библиотеката ***math***

Свързващ редактор

Понякога една програма е твърде дълга за да се запише като един файл, а може и отделните и части да се пишат от различни програмисти. Така може отделните и части да са оформени като самостоятелни файлове.

Това налага съществуването на свързващ редактор, който да обединява компилираните части да се обединят в една изпълнима програма, Друга роля на свързващия редактор е да свърже с програмата всички необходими библиотеки.

Интегрираща обвивка

Програмите на средата могат да бъдат обединени от една програма, която програмиста да извика и, от която да се обръща към всички останали. Тази програма ще наричаме **обвивка на средата**.

Средите, в които всички програми, необходими за разработването на нови програми са интегрирани в обвивка се наричат **интегрирани среди за програмиране / разработване**