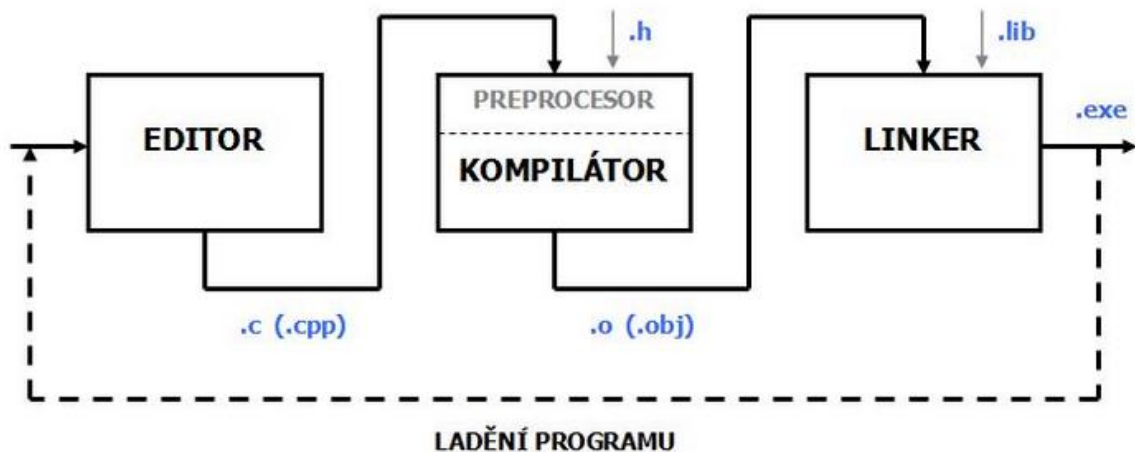


1.Programovací jazyk C

- Vývojové prostředí Dev C++, compiler a linker
- Datové typy
 - Definice vs. inicializace
- Přetypování
- Standardní vstupy a výstupy
- Řídící struktury
 - Větvení
 - Cykly
- Práce s polem
 - Jednorozměrné, dvourozměrné
 - Načtení, výpis, manipulace s prvky,...
- Algoritmy třídění pole
- Funkce
 - Popis a použití
 - Definice, tělo, prototyp
 - Bez parametru, s parametrem,...
- Struktury a ukazatele
- Práce se soubory
 - Typy souborů
 - Deklarace proměnných pro práci se souborem
 - Přístupové funkce
- Knihovny



1. Vývojové prostředí Red Panda C++, compiler a linker

- **Red Panda C++** je integrované vývojové prostředí (IDE) pro programovací jazyk C/C++/GNU Assembly. Poskytuje uživateli programovací prostředí s podobnou uživatelskou přívětivostí jako VS code nebo CLion, ale s mnohem menší náročností na systém.
- **Kompilátor** (překladač) soubor .c nebo .cpp nejdříve zpracuje preprocesorem, který zdrojový kód částečně upraví – vloží hlavičkové soubory, odstraní komentáře apod. Poté kompilátor přeloží příkazy, čímž vznikne tzv. relativní kód (s příponou .obj). V případě, že se v programu vyskytnou formální chyby, je nutné je opravit a kód znovu zkompileovat.
- **Linker** (sestavovací program) je poslední fází, kde se propojí relativní kód a knihovní soubory (.lib). Výsledkem je spustitelný soubor (ve Windows s příponou .exe).

2. Datové typy

- Datové typy v jazyce C určují typy dat, které můžeme používat v programu. Říkají kompilátoru, jak má s danou proměnou pracovat, a proto je musíme dodržovat. Patří sem například celá čísla (int, long), desetinná čísla (float, double), znaky (char), typ žádné hodnoty (void) a další.
- **Definice vs. inicializace:**
 - Konstantní proměnná musí být definována jinými slovy přiřazenou hodnotou ve stejném příkazu, ve kterém je deklarována. Deklarace předdefinovaného typu, jako int je automaticky definice, protože kompilátor ví, kolik místa pro něj má přidělit.

3. Přetypování

- Přetypování (také známé jako typecasting) je proces převodu hodnoty z jednoho datového typu na jiný.

4. Standardní vstupy a výstupy

%c	znak
%d, %i	celé desítkové číslo
%f	reálné číslo (float)

- Standardní vstupy (stdin) a výstupy (stdout) jsou předdefinované proudy vstupu a výstupu v jazyce C, které umožňují interakci s uživatelem pomocí funkcí jako printf() a scanf() nebo.

5. Řídící struktury

- Řídící struktury v jazyce C určují tok programu. Patří sem větvení (if), cykly (for, while, do-while) a skoky (break, continue, return).
- **Větvení:**
 - Větvení umožňuje programu rozhodovat se na základě podmínek a provádět různé části kódu v závislosti na jejich splnění.
- **Cykly:**
 - Cykly umožňují opakování části kódu, dokud je podmínka splněna.

6. Práce s polem

- Pole jsou kolekce prvků stejného typu, uložených v paměti za sebou. V jazyce C můžeme pracovat s jednorozměrnými a dvourozměrnými poli.
- **Jednorozměrné, dvourozměrné:**
 - Jednorozměrná pole jsou pole s jedním rozměrem, zatímco dvourozměrná pole mají dva rozměry a jsou organizována jako matice.
- **Načtení, výpis, manipulace s prvky:**
 - Manipulace s prvky pole zahrnuje načítání hodnot do pole, výpis hodnot z pole a provádění operací s hodnotami v poli.

7. Algoritmy třídění pole

- Algoritmy třídění pole jsou algoritmy, které uspořádají prvky pole do určitého pořadí. Patří sem například algoritmy jako Bubble Sort, Selection Sort nebo Quick Sort, pro který existuje funkce z knihovny stdlib.h (qsort).

8. Funkce

- Funkce jsou bloky kódu, které mohou být volány z jiných částí programu. Mohou být bez parametru, s parametrem a mohou vracet hodnotu. V případě použití void* může být návratová hodnota libovolného typu
- **Popis a použití:**
 - Funkce umožňují organizovat kód do menších, snadněji spravovatelných částí a opakovaně používat určité úseky kódu.
- **Definice, tělo, prototyp:**
 - Funkce se skládá z deklarace (prototypu) a definice (těla). Prototyp obsahuje název funkce, datový typ návratové hodnoty a datové typy parametrů, zatímco tělo obsahuje samotný kód funkce.
- **Bez parametru, s parametrem:**
 - Funkce může mít parametry, které přijímá a zpracovává, nebo nemusí mít žádné parametry.

9. Struktura a ukazatelé

- Funkce mohou pracovat s datovými strukturami a ukazateli, což umožňuje efektivní manipulaci s daty a alokaci paměti.

10. Práce se soubory

- Práce se soubory v jazyce C umožňuje čtení a zápis dat ze souborů na disku.
- **Typy souborů:**
 - Soubory mohou být textové nebo binární. Textové soubory obsahují lidsky čitelný text, zatímco binární soubory obsahují datové struktury v binárním formátu.
- **Deklarace proměnných pro práci se souborem:**
 - Pro práci se souborem je třeba deklarovat proměnnou typu `FILE`, která představuje ukazatel na soubor.
- **Přístupové funkce:**
 - K práci se soubory slouží různé přístupové funkce, jako například `fopen()` pro otevření souboru, `fclose()` pro uzavření souboru, `fread()` pro čtení ze souboru a `fwrite()` pro zápis do souboru.

11. Knihovny

- Knihovny v jazyce C obsahují předdefinované funkce, které lze použít v programu. Patří sem standardní knihovny jako `stdio.h`, `stdlib.h`, `math.h`, `string.h`, `ctype.h` a další.