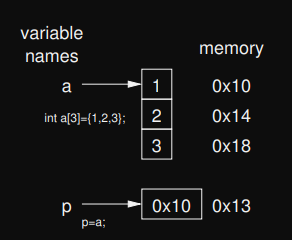
Pointery

* Pointery jsou uložené v zásobníku, ale data, na které ukazují jsou uložené v haldě. Při vytvoření pointeru se alokuje místo v zásobníku a zapíše se do něj adresa v haldě, na kterou ukazuje.
* 

# Využití pointerů:

## Dynamické alokování paměti (malloc)

int \*pointer;

Pointer = (int \*)malloc(sizeof(int)) //nastaví adresu na kterou pointer ukazuje na velikost 1 intu (4B)

\*pointer = 42; // na adresu uloží číslo 42

free(pointer); // uvolnění paměti aby nedošlo k úniku

## Práce s poli (viz otázka 12)

## Funkce s pointery (vrácení více hodnot z funkce)

Void vymena(int \*a, int \*b) {

Int temp = \*a;

\*a = \*b;

\*b = temp;

}

Int main() {

….

vymena(&x,&y); // do proměnných x a y se uloží hodnoty a,b z funkce vymena()

}

## Manipulace s textovými řetězci

Char str[] = “Hello world!”;

Char \* pointer = str;

while(\*pointer != ‘\0’) {

printf(“%c”, \*pointer);

pointer++;

}

## Práce s dvourozměrnými poli (viz otázka 13)

# Práce s pointery:

1. Deklarace:

Int \*ptr;

1. Přiřazení adresy

Int num = 10;

ptr = &num;

1. Vypisování hodnoty

printf(“Hodnota na adrese: %d \n”, \*ptr);

1. Vypisování adresy, na kterou ukazuje pointer

printf(“Adresa pointeru: %p“, &ptr);

Obsah obrázku text, snímek obrazovky, Písmo, účtenka

Popis byl vytvořen automaticky

## Pointerová aritmetika

p+= 2; //posunutí o 2 pole v paměti

p--; //posunutí o pole zpět v paměti