

Auszuarbeiten bis 18.03.21

1. Zeigermanipulationen (1 + 2 + 2 Punkte)

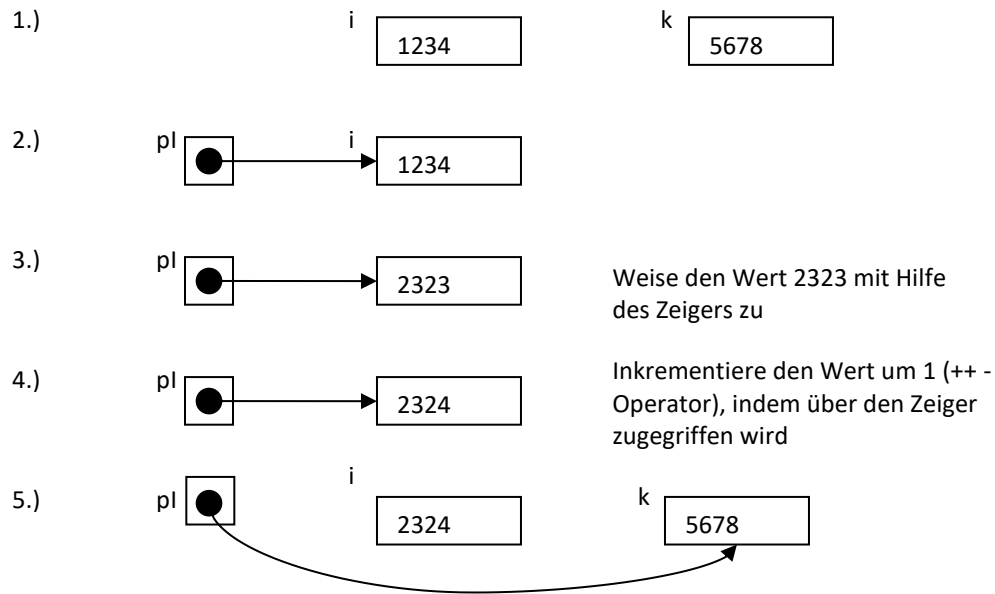
a) Was ist ein Zeiger? Wie unterscheidet sich eine Zeigervariable von einer „herkömmlichen“ Variable?

b) Beschreiben Sie für den folgenden Code Zeile für Zeile, was gemacht wird,

z.B.: `printf("%d", *pA)` „Der Pointer pA wird mittels *-Operator dereferenziert und ausgegeben, erwartetes Ergebnis ist: ...“

```
/* 01 */      int i = 5;
/* 02 */      int j = 6;
/* 03 */      int t = 0;
/* 04 */      int* pI = NULL;
/* 05 */      int* pJ = &j;
/* 06 */      float* pF = (float*) &i;
/* 07 */      pI = &i;
/* 08 */      printf("%i, %i", *pI, *pJ);
/* 09 */      t = *pI;
/* 10 */      *pI = *pJ;
/* 11 */      *pJ = t;
/* 12 */      printf("%i, %i", *pI, *pJ);
/* 13 */      pJ = pI;
/* 14 */      printf("%i, %i", *pI, *pJ);
/* 15 */      *pJ = 10;
/* 16 */      printf("%i, %i", *pI, *pJ);
```

c) Schreiben Sie ein Programm, das die folgende Grafik Schritt für Schritt implementiert mit einer Ausgabe nach jedem Schritt:



2. Tokenizer (5 + 5 Punkte)

Entwickeln Sie eine Funktion `tokenize`, die innerhalb eines Satzes die einzelnen Worte identifiziert. Ein Satz ist eine Zeichenkette, die aus Groß- und Kleinbuchstaben und Leerzeichen zur Worttrennung besteht.

Entwickeln Sie zwei Versionen der Funktion. Die erste Version liefert die Indices (Feld von Integerwerten), die zweite die Adressen (Feld von `char`-Zeigern), an denen die vorkommenden Worte anfangen, zurück.

Beispiel:

Text	D	i	e	s		i	s	t		e	i	n		P	r	o	b	e	t	e	x	t	\0
Index	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	
Adr	0x12			0x17																	

Resultat der 1. Version: 0, 5,...

Resultat der 2. Version: 0x12, 0x17, ...

Geben Sie die Resultate im Testtreiber für verschiedene Sätze aus!

3. Zeichenkette kopieren mit dynamischem Speicher (5 Punkte)

Implementieren Sie eine C-Funktion, die als Parameter eine Zeichenkette erhält, diese in einen neuen, dynamisch angelegten Speicherbereich kopiert und die duplizierte Zeichenkette als Rückgabewert zurückliefert.

```
char* strDuplicate(const char * pSrc);
```

Achten Sie auf ordnungsgemäße Allokation und Deallokation des dynamisch angelegten Speichers!

1 Zeigermanipulationen

- a) Eine Zeigervariable (Pointer) beinhaltet eine Adresse die auf eine Speicheradresse im Speicher zeigt. Der Wert des Pointers ist eine hex zahl, was die speicherstelle representiert. Wird ohne Derefernzierungsoperator (*) gearbeitet, wird auf die Adresse zugegriffen. Mit Derefernzierungsoperator wird auf den Wert, der bei der Speicheradresse des Pointers liegt, zugegriffen.
- b) **01:** Es wird eine Integer Variable ,i' erstellt und der Wert ,5' zugewiesen.
02: Es wird eine Integer Variable ,j' erstellt und der Wert ,6' zugewiesen.
03: Es wird eine Integer Variable ,t' erstellt und der Wert ,0' zugewiesen.
04: Es wird ein Integer Pointer ,pl' erstellt und die Adresse ,Null' zugewiesen.
05: Es wird ein Integer Pointer ,pj' erstellt und die Adresse von ,j', über den ,&' operator, zugewiesen.
06: Es wird ein Float Pointer ,pf' erstellt und die Adresse von ,i', über den ,&' operator und typecasting, zugewiesen.
07: Dem Pointer ,pl' wird die Adresse von ,i' zugewiesen.
08: Die Werte der beiden Pointer Variablen, ,pl' und ,pj', werden ausgegeben.
09: Der Variable ,t' wird der Wert von dem Pointer ,pl', über den Derefernzierungsoperator (*), zugewiesen.
10: Der Wert des Pointers ,pj' wird dem Wert des Pointers ,pl' zugewiesen.
11: Der Wert der Variable ,t' wird dem Wert des Pointers ,pj' zugewiesen.
12: Die Werte der beiden Pointer Variablen, ,pl' und ,pj', werden ausgegeben.
13: Die Adresse des Pointers ,pl' wird der Adresse des Pointers ,pj' zugewiesen.
14: Die Werte der beiden Pointer Variablen, ,pl' und ,pj', werden ausgegeben.
15: Der Wert ,10' wird dem Wert des Pointers ,pj' zugewiesen.
16: Die Werte der beiden Pointer Variablen, ,pl' und ,pj', werden ausgegeben.
- c) C-Program code: Siehe beigelegtes main.c file.

Ausgabe des C-Programms:

```
#### pointerManipulations #####
Adresse pl = 00AFF708   Wert von pl = 1234       Wert von i = 1234       Wert von k = 5678
Adresse pl = 00AFF708   Wert von pl = 2323       Wert von i = 2323       Wert von k = 5678
Adresse pl = 00AFF708   Wert von pl = 2324       Wert von i = 2324       Wert von k = 5678
Adresse pl = 00AFF70C   Wert von pl = 5678       Wert von i = 2324       Wert von k = 5678
```

2 Tokenizer

Testfälle:

Text	Dies	ist	ein	Probetext
Indices	0	5	9	13
Addresses	0x010FF7E0	0x010FF7E5	0x010FF7E9	0x010FF7ED

Text	Dies
Indices	0
Addresses	0x012FFB00

Text
Indices
Addresses

C-Program code & Test Ausgaben --> Siehe beigelegtes .c file.

3 Zeichenkette kopieren mit dynamischem Speicher

Testfälle:

Text Dies ist ein Probetext

Text Dies

Text

C-Program code & Test Ausgaben --> Siehe beigelegtes .c file.