

4. praktikum

IAG0581 Programmierimine I

Muutujatüübid

- **char** – tähemärk (character) 1 bait [0, 255]
- **int** – täisarv (integer) 2 baiti [-32767, +32767]
- **float** – ujukomaarv (floating point number) 4 baiti [1.2E-38, 3.4E+38]
- **double** – ujukomaarv (double precision float) 8 baiti [2.3E-308, 1.7E+308]

Muutujate deklareerimine

- Muutuja seostamine muutujatüübiga ja mälu eraldamine

```
int a;
```

- Muutujale väärtuse omistamine

```
a = 25;
```

- Näited

```
int number = 5;
```

```
float average = 0.25;
```

```
double median = 0.0525;
```

```
char letter = 'a';
```

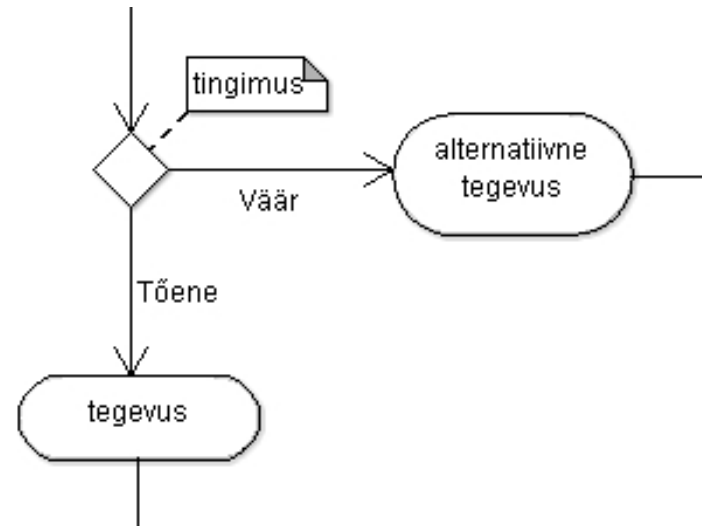
Operaatorid

Operaator	Kirjeldus	Näide
=	Omistamine	int a = 10;
+	Liitmine	a = 10 + 1;
-	Lahutamine	int b = a - 2;
*	Korrutamine	int c = a * b;
/	Jagamine	float d = a/5;
%	Jääk (Modulus/mod)	b = 14 % 10; // tulemus 4
++	Täisarvu suurendamine ühe võrra	i++; // i = i + 1;
--	Täisarvu vähendamine ühe võrra	j--; // j = j - 1;
+=	Arvu suurendamine antud väärtuse võrra	k+=5; // k = k + 5;
-=	Arvu vähendamine antud väärtuse võrra	k-=5; // k = k - 5;

Tingimuslause

```
if (tingimus) {  
    tegevus;  
} else {  
    alternatiivne tegevus;  
}
```

```
if (tingimus1) {  
    tegevus;  
} else if (tingimus2) {  
    tegevus;  
} else {  
    alternatiivne tegevus;  
}
```



Tingimuse operaatorid

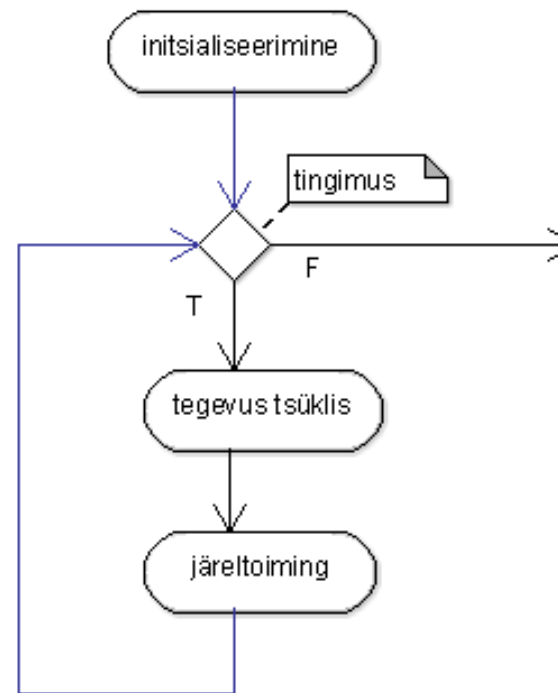
Operaator	Kirjeldus	Näide
==	Võrdne	a == b
!=	Mitte võrdne	a != c
< >	Väiksem-kui; suurem-kui	a > c
<= >=	Väiksem-võrdne-kui; suurem-võrdne-kui	a <= b
&&	JA operaator. Mõlemad tingimused peavad tõesed olema.	(a == b) && (a >= c)
	VÕI operaator. Üks tingimustest peab tõene olema.	(a == b) (a >= c)

Tsükkel

- Tsükkel on korduv tegevus
- Tsüklid jagunevad eelkontrolliga ja järelkontrolliga tsükliteks (millal kontrollitakse tingimust?)
- Eelkontrolliga:
 - while
 - for
- Järelkontrolliga:
 - do while

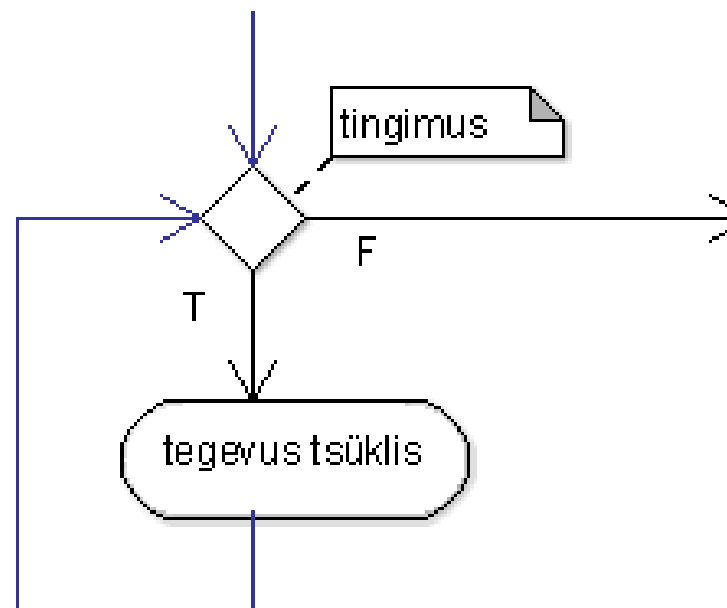
For tsükkel

```
for (initsialiseerimine; tingimus; järeltoiming) {  
    tegevus tsükliis;  
}
```



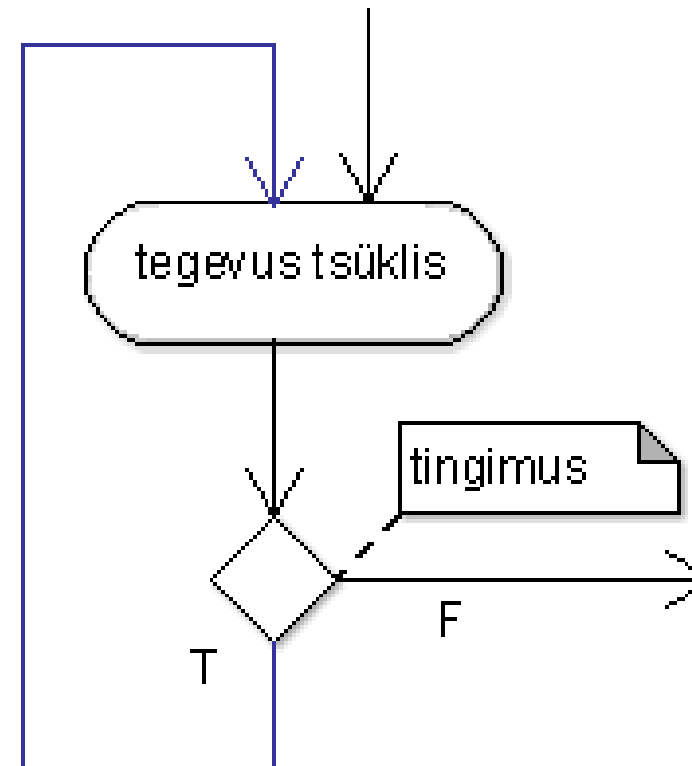
While tsükkel

```
while (tingimus) {  
    tegevus tsükli;  
}
```



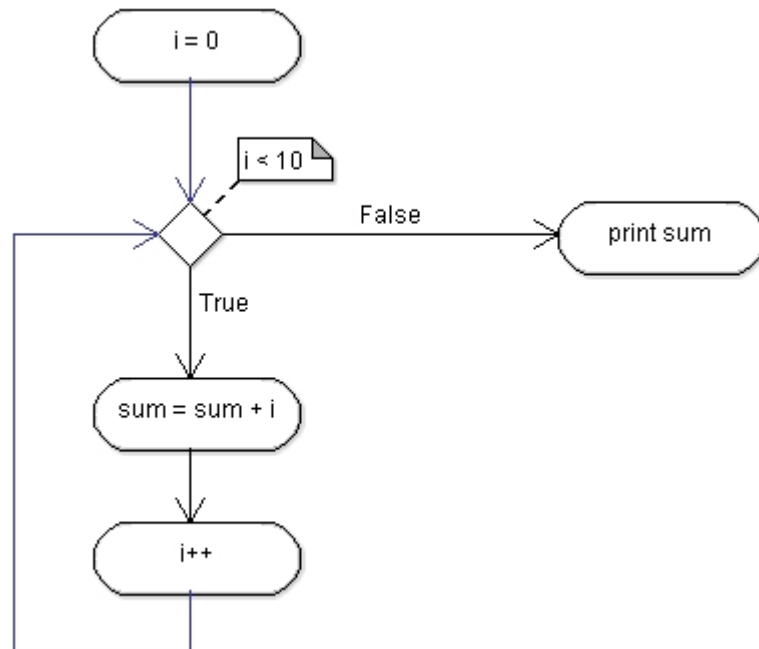
Do-while tsükkel

```
do {  
    tegevus tsükliis;  
} while (tingimus);
```



Tsükli näide

- Arvude 1 kuni 10 summa



```
int sum = 0;
int i;
for (i=0; i<10; i++) {
    sum = sum + i;
}
printf("%d", sum);
```

```
int sum = 0;
int i = 0;
while (i<10) {
    sum = sum + i;
    i++;
}
printf("%d", sum);
```

C standard input output teek

- Teegi lisamine

```
#include <stdio.h>
```

- **scanf** - Klaviatuurilt sisestamine

- Näide: `int arv;`
`scanf("%d", &arv);`

- Muutujanime ees peab olema ampersand (&). Viitab mäluaadressile.

- **printf** – Teksti väljastamine terminali

- Näide: `int number = 5;`
`printf("Number: %d", number);`

- Muutujanime ees ampersandi pole. Viitame väärtusele.

printf ja scanf formaat

<u>Muutujatüüp</u>	<u>Selgitus</u>	<u>Formaat</u>	<u>Näide</u>
• Integer	täisarv	%d	5
• Float/double	komaga arv	%f	3.25
• String	sõne / tekstijada	%s	"Tere 123"

- Täpsemalt:

<http://www.cplusplus.com/reference/cstdio/printf/>

Kompileerimine

- Kood: `#include <stdio.h>`

```
int main(void) {  
    char tekst[32];  
    printf("Sisesta tekst: ");  
    scanf("%s", &tekst);  
    printf("Tere %s\n", tekst);  
  
    return 0;  
}
```

- Salvestada nimega hello.c ja terminalis sisestada:
`gcc hello.c -o hello`

Kommenteeritud kood

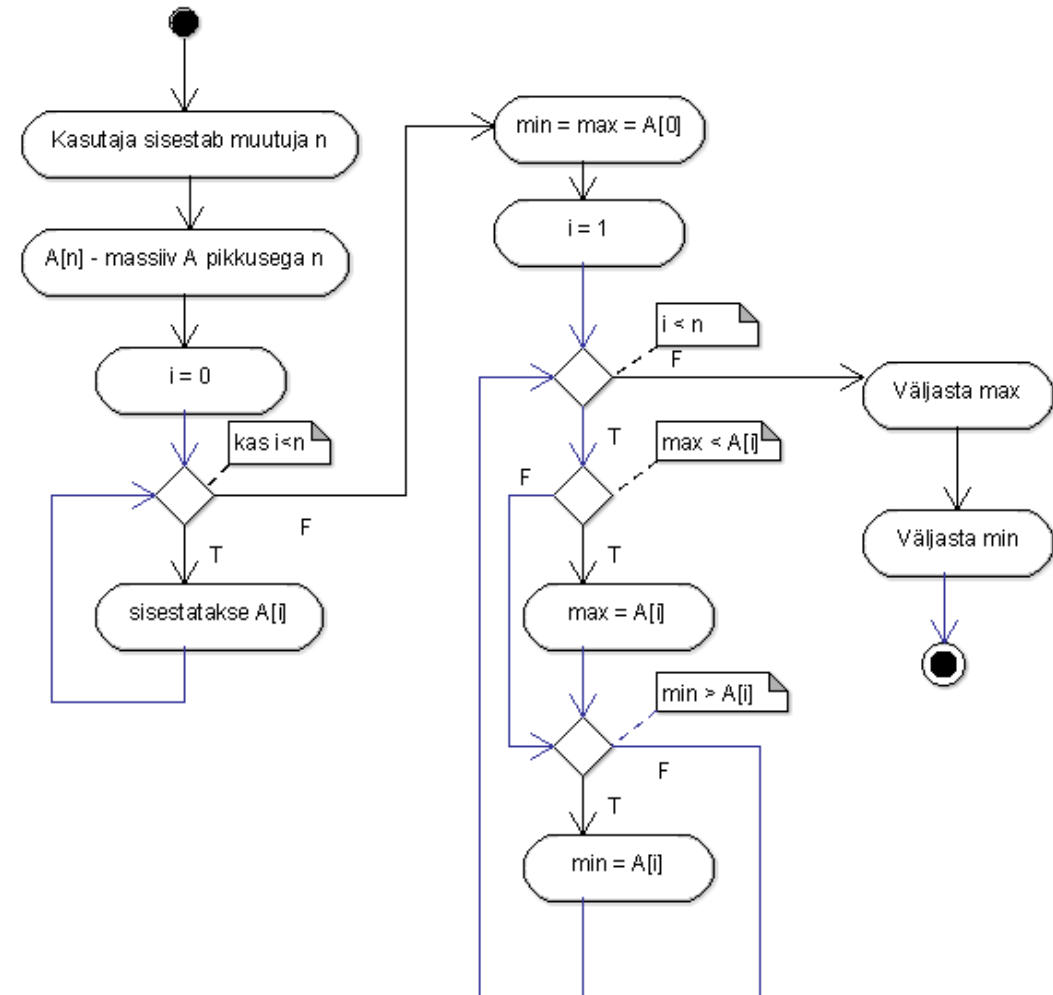
```
#include <stdio.h> // lisa teek C standard input output klaviatuurilt lugemiseks
ja terminali väljastamiseks

int main(void) { // peafunktsioon
    char tekst[32]; // muutuja: tähemärkide massiiv ehk string, pikkusega 32
    printf("Sisesta tekst: "); // Väljastame ekraanile
    scanf("%s", &tekst); // loeme klaviatuurilt muutuja väärtuse
    printf("Hello %s\n", tekst); // väljastame muutuja väärtuse ekraanile

    return 0; // tagastame programmi töö lõppedes 0, mis tähendab, et programm
lõpetas oma töö ootuspäraselt
}
```

Näidisülesanne

- Luua programm, mis laseks kasutajal sisestada n täisarvu ning leiaks nendi hulgast maksimumi ja miinimumi.



Tunniülesanne

- Kirjutada eelmises tunnis loodud mullisordi algoritmi põhjal programm, mis sorteeriks kasutaja poolt sisestatud arvud ja väljastaks need sorteeritult ekraanile.
- Kui on liiga lihtne, implementeerige mullisordi asemel Shell sort.
- Valmis töö laadida GIT-i

- <http://www.blue.pri.ee/ttu/koodimisstiil/>

Lisaülesanne

- Luua programm koos algoritmi kirjeldusega, mis leiaks etteantud kuupäeva põhjal, milline nädalapäev sellele kuupäevale vastab.

Näiteks:

sisend: 22.09.2015

väljund: Teisipäev

- https://en.wikipedia.org/wiki/Determination_of_the_day_of_the_week