



# UiT Norges arktiske universitet

<b>Skoleeksamen i:</b>	INF-1100 Innføring i programmering og datamaskiners virkemåte
<b>Dato og tidspunkt:</b>	2022-02-27 Kl 09:00 - 13:00
<b>Målform:</b>	Bokmål
<b>Fagansvarlig (navn og telefonnr.):</b>	Edvard Pedersen, tlf. 40458598
<b>Antall sider:</b>	4
<b>Tillatte hjelpe midler:</b>	Ingen
<b>Vil det bli gått runde av fagansvarlig? Hvis ja, ca. når? :</b>	11:00
<b>Eventuell annen informasjon:</b>	

Les oppgaveteksten grundig og disponer tiden slik at du får tid til å svare på alle oppgavene. I noen oppgaver kan det være nødvendig å tolke oppgaveteksten ved å gjøre noen antagelser - gjør i så fall rede for hvilke antagelser du har gjort, men pass på å ikke gjøre antagelser som trivialiserer oppgaven. Der du skal utvikle eller beskrive en algoritme anbefales det at du først beskriver algoritmen på et høyt abstraksjonsnivå, f.eks. med figurer, før du går videre med detaljer og eventuell pseudokode. Der du blir spurta om å implementere noe, så kan du skrive i C-lignende pseudokode, skrivefeil eller syntaxfeil trekker ikke ned. Husk også at du kan referere tilbake til funksjoner du tidligere har definert.

## Oppgave 1 - 25%

Gi en kort beskrivelse av von Neumann modellen og instruksjonssyklusen. Beskrivelsen bør omfatte de ulike komponentene i modellen og hvordan disse interagerer med hverandre.

## Oppgave 2 - 25%

Gitt følgende funksjon:

```
char *ukjent(char *c) {
    char *d = malloc(sizeof(char) * 100);
    char *r = d;
    char *e = c;
    int i = 0;
    while(*e && i < 100) {
        i++;
        e++;
    }
    e--;
    while(i > 0) {
        *d = *e;
        d++;
        i--;
        e--;
    }
    return r;
}
```

- a) Hvilken verdi vil funksjonen *ukjent* ovenfor returnere ved følgende kall:
  - (a) *ukjent("Hello world!")*
- b) Hva er poenget med uttrykket *e--*; mellom while-løkkene?.
- c) Hvorfor returneres *r* og ikke *d*?

## Oppgave 3 - 25%

Gitt følgende strukter for å representere kort fra en kortstokk og en hånd med kort:

```
typedef struct card card_t;
struct card {
    int value;
    int type_of_card;
};

typedef struct hand hand_t;
struct image {
    int num_cards;
    card_t *cards;
};
```

- a) Skriv en funksjon *deal* som lager *num\_hands* hender med *num\_cards* kort i hver hånd. (merk, du kan anta det finnes en *rand()*-funksjon som kan brukes til å velge tilfeldige kort, og du repeterete kort går fint)

```
hand_t *deal(int num_hands, int num_cards)
```

- b) Skriv en funksjon *play* som fjerner ett kort fra den gitte hånden, og returnerer det. Det skal returnere NULL hvis kortet ikke finnes på hånden.

```
card_t *play(hand_t *hand, int value, int type_of_card)
```

## Oppgave 4 - 25%

Gitt følgende strukturer og funksjoner:

```
typedef struct listnode listnode_t;
struct listnode {
    listnode_t *next;
    void *item;
}

typedef struct list list_t;
struct list {
    listnode_t *head;
    int numitems;
}

// Checks an item against a filter
// Returns 1 if item passes the filter
// Returns 0 if item does not pass the filter
int filter(void *item);
```

- a) Lag en funksjon *list\_create\_iterator* som lager en iterator for å gå gjennom listen, og som også tar inn en funksjonspeker til en filter-funksjon (se eksempelet på en filter-funksjon over), du må også lage en *list\_iterator\_t* struktur.

```
typedef struct list_iterator list_iterator_t;
list_iterator_t *list_create_iterator(list_t *list, filterfunc_t filter)
```

- b) Lag funksjonen *list\_next*, som returnerer neste element i listen, gitt en iterator, og *list\_next\_filtered* som returnerer neste element som går gjennom filteret.

```
void *list_next(list_iterator_t *iter)
void *list_next_filtered(list_iterator_t *iter)
```