DATO: 7 desember 1995

HØGSKOLEN I STAVANGER
AVDELING FOR TEKNISK-NATURVITENSKAPELIGE FAG
EKSAMEN I: TE495 Programmering 2
LENGDE PÅ EKSAMEN: 5 timer, 09.00-14.00
TILLATTE HJELPEMIDLER: Alle skrevne; kalkulator
OPPGAVESETTET BESTÅR AV: 3 sider

Råd for oppgaveløsning:

Les hele oppgaveteksten før du begynner å svare på de enkelte oppgavene. Dersom du finner at teksten ikke gir deg tilstrekkelig informasjon, kan du gjøre dine egne forutsetninger. Du må i så fall presisere disse forutsetningene i besvarelsen.

Vær kort og konsis i besvarelsene. Dersom du får knapp tid er det bedre å skrive pseudokode/skisser for de oppgaver du ikke får tid til fremfor å la oppgavene være blanke.

Ved besvarelse av programmeringsoppgavene er det vesentlig at du er nøye med dokumentasjonen. Skriv tilstrekkelig med kommentarer.

Oppgave 1

(a) **[5%]** Hva er forskjellen mellom en klasse og et objekt i C++?

(b) **[5%]** Hva er forskjellen mellom en *constructor* og en *tilordningsoperator*? Gi generelle eksempler på begge.

(c) **[5%]** For en klasse kan vi lage operatorer både som medlemsfunksjoner og globale funksjoner. Forklar *kort* hvordan, samt hvorfor det noen ganger er behov for globale funksjoner for dette.

(d) **[5%]** Hva er *arv* i C++?

(e) **[5%]** Hva er polymorfi? Hvordan implementerer man dette i C++?

(f) **[5%]** Hvilke av følgende er ulovlige objektnavn i C++? Forklar hvorfor.

- `fire`
- `4`
- `ett objekt`
- `supercallifragilisticexpialidocious`
- `output_temperature3`

(g) **[5%]** Gi C++-ekvivalentene til følgende algebraiske uttrykk; `p`, `q`, `x` og `y` er alle av typen `double`.

- $(3p + q)(p - 2q)$
- $\frac{1}{x+y} + \frac{1}{x-y}$
- $x^4 + x^3 + x^2 + x + 1$
- $1 + \frac{1}{1+1/(1+x)}$
- $\sin^2(x/y)$

Oppgave 2

(a) **[15%]** Skriv et program som leser inn flyttall fra en fil av ukjent størrelse og skriver ut antallet, det minste, det største samt middelverdien av alle tallene til skjermen. Middelverdien skal skrives med 1 desimal.

(b) **[15%]** Et primtall er et heltall som ikke kan deles (uten rest) med annet enn 1 og seg selv. Lag en funksjon som returnerer TRUE (1) hvis inn-parameteren er et primtall, FALSE (0) ellers. Besvarelsen skal inneholde en algoritmebeskrivelse (flytskjema eller lignende) som forklarer hvordan du har tenkt.

Oppgave 3

Et system skal lages for å holde rede på alle dyrene man har i en zoologisk hage. For hvert dyr skal man holde orden på:

- Hva slags dyr det er
- Dyrets navn
- Når dyret ble født (klekket)
- Dyrets foreldre (hvis kjent)

Samtlige dyr skal registreres med et serienummer; dette kan brukes til å finne et dyrs foreldre.

Følgende utgangspunkt er gitt (i filen `dyr.h`):

```
#include "dato.h"

enum Familie {Pattedyr, Pungdyr, Fugl, Fisk, Reptil};      // etc...
enum Rase    {Ulv, Hund, Leopard, Kenguru, Pungrotte, Pingvin, Albatross,
              Klapperslange, Krokodille};

class Dyr {
  char *Navn;
  unsigned Serienr;
  unsigned Far, Mor;
  Dato Fdato;
public:
  Dyr(char *navnet, unsigned nummeret, char *fdatoen,
      unsigned faren=0, unsigned moren=0);
  ~Dyr();
  unsigned FinnFar();
  unsigned FinnMor();
  virtual Rase FinnRase() = 0;
  virtual Familie FinnFamilie() = 0;
  virtual void Skriv(ostream&);
};
```

En antar her at en klasse `Dato` er deklarert i den inkluderte filen. Du trenger ikke å definere denne klassen; anta at den har en constructor

```
Dato::Dato(char *datoen); // streng på formen "31/12/94"
```

samt en utskriftsfunksjon

```
Dato::Skriv(ostream&);    // skriver tilsvarende streng til gitt ostream
```

(a) **[5%]** Beskriv (tegn) et arvehierarki som går fra den generelle klassen `Dyr` via familene til de enkelte raser. Bruk gjerne navnene fra `enum`-eksemplet ovenfor. For de enkelte klassene skal du beskrive hva som bør være med av ekstra data samt medlemsfunksjoner. Prøv å legge begge deler så langt *opp* i hierarkiet som mulig, og ikke ta med mer enn nødvendig. Les også deloppgave (**f**) før du lager strukturen din!

(b) **[5%]** `Ulv` kan være en underklasse av `Pattedyr`, som igjen arver fra `Dyr`. Utskriftsfunksjonen til en ulv kan resultere i noe slikt som:

```
Ulv, Pattedyr: Fido. Serienr. 112, Født 31/12/94, mor:110, far:111
```

Lag `Skriv()`-funksjoner for de gjeldende klasser slik at dette faktisk blir tilfellet når man vil skrive ut informasjon om en ulv.

(c) **[5%]** Lag passende constructorer for `Dyr`, `Pattedyr` og `Ulv`.

(d) **[5%]** Lag destructoren for `Dyr`. Trenger man å lage destructorer for `Pattedyr` og/eller `Ulv`? Begrunn svaret.

(e) **[5%]** Lag et lite hovedprogram som registrerer tre ulver (*Fido* er hvalp av *Varg* og *Tass*), en krokodille og en pingvin, samt skriver ut registrert informasjon om disse. Bruk polymorfi.

(f) **[5%]** Hvordan kan man enkelt telle opp antall dyr av en gitt rase (eller familie) med strukturen du har valgt?

(g) **[5%]** I basisklassen er mor og far referert ved hjelp av deres serienummer. En bedre metode kunne kanskje være å bruke et par pekere her. Beskriv hvordan du kunne tenke deg dette gjort.