



UNIVERSITETET I BERGEN

KANDIDAT

243

PRØVE

INF122 0 Funksjonell programmering

Emnekode	INF122
Vurderingsform	Skriftlig eksamen
Starttid	09.12.2024 14:00
Sluttid	09.12.2024 17:00
Sensurfrist	--
PDF opprettet	04.01.2025 15:02

Eksamensinformasjon

Oppgave	Tittel	Oppgavetype
---------	--------	-------------

Eksamensoppgaver

Oppgave	Tittel	Oppgavetype
---------	--------	-------------

1	Oppgaver	Muntlig
---	----------	---------

1 Oppgaver

Se på vedlagt PDF for eksamensspørsmålene. Du kan justere størrelsen på PDF-en ved å klikke på de tre prikkene og dra linjen.

Eksamensspørsmålene finnes også ved å klikke på knappen under navigasjonslinjen, hvis du foretrekker å ha PDF-en i en egen fane.

Svarene dine skrives på papir, kalt Scantron-ark. Be eksamensvakten om disse arkene dersom du ikke ha fått disse allerede. Det er viktig at du refererer til hvilke spørsmål du svarer på. F.eks.: 1.1, 1.2, 2.1, 2.2 osv.

Hvordan fylle ut scantron-ark:

En kode vil være tilgjengelig under denne boksen. Koden er unik for hvert spørsmål per student. Du må derfor merke arket du har skrevet på med spørsmålsnummeret og koden du har svart på under eksamen.

I de 15 minuttene etter eksamenstiden kan du fylle ut annen forespurt informasjon øverst på Scantron-arkene: dato, kandidatnummer, emnekode, antall sider osv. Kandidatnummeret ditt finner du i eksamenssystemet.

Spør eksamensvakten hvis du har problemer med å finne koden eller kandidatnummeret ditt. Når du er ferdig med eksamen, skal arkene leveres samlet, i den rekkefølgen du har nummert disse, til hovedvakten i lokalet. Disse vil bli skannet inn og lastet opp til Inspira i den rekkefølgen du har levert disse inn.

Knytte håndtegninger til denne oppgaven?

Bruk følgende kode:

3 2 3 5 8 7 3

i Fyll inn oppgavekode og emneinformasjon på alle skissearkene

Fill out question code and test information on every sheet

Oppgavekode
Question codeDato
DateEmnekode
Subject codeKandidatnummer
Candidate numberOppgavenummer
Question numberSidetall
Page number

3	2	3	5	8	7	3
0	0	0	0	0	0	0
1	1	1	1	1	1	1
2	2	2	2	2	2	2
3	3	3	3	3	3	3
4	4	4	4	4	4	4
5	5	5	5	5	5	5
6	6	6	6	6	6	6
7	7	7	7	7	7	7
8	8	8	8	8	8	8
9	9	9	9	9	9	9

12/9/2024

INF122

243

1

1/4 av/of

Tegneområde Drawing area

1.1) (d) [2,4,6,8,10]

1.2) $a = ["ab", "cd", "", "efg"]$
 $b = ['a', 'b', 'c', 'd', 'e', 'f', 'g']$
 $c = \text{Error (evg loop)}$
 $d = "abcde f g"$

1.3) $a = [0, 1, 1, 1, 1]$
 $b = [0, 0, 0, 0, 0]$
 $c = [1, 2, 2, 3, 2]$
 $d = [0, 1, 2, 3, 4]$

1.4) (b) $(a \rightarrow b) \rightarrow a \rightarrow b$

Fyll inn oppgavekode og emneinformasjon på alle skissearkene

Fill out question code and test information on every sheet

Oppgavekode
Question codeDato
DateEmnekode
Subject codeKandidatnummer
Candidate numberOppgavenummer
Question numberSidetall
Page number

3	2	3	5	8	7	3
0	0	0	0	0	0	0
1	1	1	1	1	1	1
2	2	2	2	2	2	2
3	3	3	3	3	3	3
4	4	4	4	4	4	4
5	5	5	5	5	5	5
6	6	6	6	6	6	6
7	7	7	7	7	7	7
8	8	8	8	8	8	8
9	9	9	9	9	9	9

12/9/2024 INF122

243

2

2/4 av/of

Tegneområde Drawing area

2.1) ZipWith f xs

2.1) ZipWith _ [] = []

ZipWith _ [] = []

ZipWith f (x:xs) (y:ys) = (f x y) : ZipWith f xs ys

Zip = ZipWith (\xy → (x,y))

2.2 ~~ap f xs = foldr (f acc → f acc) f xs~~

ap f xs = foldr (f acc → f acc) x f xs

2.3 ap f xs = map (ap f) xs

i Fyll inn oppgavekode og emneinformasjon på alle skissearkene

Fill out question code and test information on every sheet

Oppgavekode
Question codeDato
DateEmnekode
Subject codeKandidatnummer
Candidate numberOppgavenummer
Question numberSidetall
Page number

3	2	3	5	8	7	3
0	0	0	0	0	0	0
1	1	1	1	1	1	1
2	2	2	2	2	2	2
3	3	3	3	3	3	3
4	4	4	4	4	4	4
5	5	5	5	5	5	5
6	6	6	6	6	6	6
7	7	7	7	7	7	7
8	8	8	8	8	8	8
9	9	9	9	9	9	9

12/9/2024 INF122

243

3

3/4 av/of

Import Data.Char

eval = do

Loop []

Loop stack = do

Input ← getLine

Process (words Input) stack

Process [] stack = do

PutStrLn \$ Show \$ head stack

Process [x] stack

1 isDigit \$ head x = do

Let newStack = xs Stack

PutStrLn newStack
Loop new Stack

1 x'elem "+" = do

Let newStack = (calculate x (take 2 Stack) % (drop 2 Stack))

PutStrLn newStack

loop newStack \$ Show

1 otherwise = Feilmelding Stack

calculate opp [x, y]

1 x == "*" = (read x) * (read y)

1 x == "+" = (read x) + (read y)

1 x == "-" = (read x) - (read y)

Feilmelding Stack = do

PutStrLn "Ikke gyldig Input"

loop stack

Tegneområde Drawing area

Process (xs) Stack
= Feilmelding Stack

Fyll inn oppgavekode og emneinformasjon på alle skissearkene

Fill out question code and test information on every sheet

Oppgavekode
Question codeDato
DateEmnekode
Subject codeKandidatnummer
Candidate numberOppgavenummer
Question numberSidetall
Page number

3	2	3	5	8	7	3
0	0	0	0	0	0	0
1	1	1	1	1	1	1
2	2	2	2	2	2	2
3	3	3	3	3	3	3
4	4	4	4	4	4	4
5	5	5	5	5	5	5
6	6	6	6	6	6	6
7	7	7	7	7	7	7
8	8	8	8	8	8	8
9	9	9	9	9	9	9

12/9/2024 INF122

243

4

4/4 av/of

Tegneområde Drawing area

4.1 fun $xy = xyy$ med hjelp av λ -uttrykk kan
~~skriv~~ skrives som $\text{fun} = \lambda xy \rightarrow xyy$

4.2 fun kan også skrives som $\text{fun} = \lambda x \rightarrow \lambda y \rightarrow xyy$

$$\begin{aligned}
 & E(\Gamma \mid \lambda x \rightarrow \lambda y \rightarrow xyy) \& t \\
 &= \{f = a \rightarrow b\} \cup E(\Gamma, x = a \mid \lambda y \rightarrow xyy \& b) \\
 &= \{f = a \rightarrow b\} \cup \{b = c \rightarrow d\} \cup E(\Gamma, x = a, y = c \mid xyy \& d)
 \end{aligned}$$

framover $\& \Gamma = x = a, y = c$

$$\begin{aligned}
 E(\Gamma \mid xyy) &= E(\Gamma \mid yy \& e) \cup E(\Gamma \mid x \& e \rightarrow d) \\
 &= E(\Gamma \mid y \& f) \cup E(\Gamma \mid y \& f \rightarrow e) \cup E(\Gamma \mid x \& e \rightarrow d) \\
 &= [f = a \rightarrow b] [b = c \rightarrow d] [f = c] [f \rightarrow e = c] [e \rightarrow d = a] \\
 &= [f = (e \rightarrow d) \rightarrow b] [b = c \rightarrow d] [f = c] [f \rightarrow e = c] \\
 &= [f = (e \rightarrow d) \rightarrow b] [b = (f \rightarrow c) \rightarrow d] [\underline{f = f \rightarrow e}]
 \end{aligned}$$

Utifra $f = f \rightarrow e$ kan vi konkludere med at fun ikke har
 noen type i Haskell