

## i **Forside**

**Institutt for datateknologi og informatikk**

**Eksamensoppgave i INFT2503 C++ for programmerere**

**Eksamensdato:** 12.05.2022

**Eksamenstid (fra-til):** 09:00 - 13:00

**Hjelpemiddelkode/Tillatte hjelpemidler:** A / Alle hjelpemidler tillatt

**Faglig kontakt under eksamen:** Ole Christian Eidheim

**Tlf.:** 90 55 16 35

**Teknisk hjelp under eksamen:** NTNU Orakel

**Tlf:** 73 59 16 00

Får du tekniske problemer underveis i eksamen, må du ta kontakt for teknisk hjelp snarest mulig, og senest innen eksamenstida løper ut/prøven stenger. Kommer du ikke gjennom umiddelbart, hold linja til du får svar.

### **ANNEN INFORMASJON**

**Ikke ha Inspira åpen i flere faner, eller vær pålogget på flere enheter, samtidig**, da dette kan medføre feil med lagring/levering av besvarelsen din.

**Skaff deg overblikk over oppgavesettet** før du begynner på besvarelsen din.

**Les oppgavene nøye**, gjør dine egne antagelser og presiser i besvarelsen hvilke forutsetninger du har lagt til grunn i tolkning/avgrensing av oppgaven. Faglig kontaktperson kan kontaktes dersom du mener det er feil eller mangler i oppgavesettet.

**Juks/plagiat:** Eksamen skal være et individuelt, selvstendig arbeid. Det er tillatt å bruke hjelpemidler, men vær obs på at du må følge eventuelle anvisningen om kildehenvisninger under. Under eksamen er det ikke tillatt å kommunisere med andre personer om oppgaven eller å distribuere utkast til svar. Slik kommunikasjon er å anse som juks.

Alle besvarelser blir kontrollert for plagiat. [Du kan lese mer om juks og plagiering på eksamen her.](#)

**Varslinger:** Hvis det oppstår behov for å gi beskjeder til kandidatene underveis i eksamen (f.eks. ved feil i oppgavesettet), vil dette bli gjort via varslinger i Inspira. Et varsel vil dukke opp som en dialogboks på skjermen i Inspira. Du kan finne igjen varselet ved å klikke på bjella øverst i høyre hjørne på skjermen. Det vil i tillegg bli sendt SMS til alle kandidater for å sikre at ingen går glipp av viktig informasjon. Ha mobiltelefonen din tilgjengelig.

### **BESVARE OG LEVERE**

**Besvare i Inspira:** Hvis oppgavesettet inneholder oppgaver som *ikke* er av typen filopplasting, skal de besvares direkte i Inspira. I Inspira lagres svarene dine automatisk hvert 15. sekund.

NB! Klipp og lim fra andre programmer frarådes, da dette kan medføre at formatering og elementer (bilder, tabeller etc.) vil kunne gå tapt.

**Automatisk innlevering:** Besvarelsen din leveres automatisk når eksamenstida er ute og prøven stenger, forutsatt at minst én oppgave er besvart. Dette skjer selv om du ikke har klikket «Lever og gå tilbake til Dashboard» på siste side i oppgavesettet. Du kan gjenåpne og redigere besvarelsen din så lenge prøven er åpen. Dersom ingen oppgaver er besvart ved prøveslutt, blir ikke besvarelsen din levert. Dette vil anses som “ikke møtt” til eksamen.

**Trekk/avbrutt eksamen:** Blir du syk under eksamen, eller av andre grunner ønsker å levere blankt/avbryte eksamen, gå til “hamburgermenyen” i øvre høyre hjørne og velg «Lever blankt». Dette kan ikke angres selv om prøven fremdeles er åpen.

**Tilgang til besvarelse:** Du finner besvarelsen din i Arkiv etter at sluttida for eksamen er passert.

## 1 Oppgave 1 (20%)

Lag nødvendig funksjon(er) slik at følgende kildekode:

```
cout << times_two(2) << endl;  
cout << times_two(2.25) << endl;  
cout << times_two("test") << endl;  
cout << times_two(vector<int>{1, 2, 3}) << endl;
```

Gir utskriften:

```
4  
4.5  
testtest  
{ 2, 4, 6 }
```

Skriv ditt svar her

1	
---	--

---

Maks poeng: 10

## 2 Oppgave 2 (40%)

a) Lag nødvendige klasse(r) slik at følgende kildekode:

```
vector<unique_ptr<ProgrammingLanguage>> programming_languages;
// Uncommenting the following line should cause compilation error:
// programming_languages.emplace_back(make_unique<ProgrammingLanguage>
// ());
programming_languages.emplace_back(make_unique<Cpp>());
programming_languages.emplace_back(make_unique<Rust>());

for (auto &programming_language : programming_languages) {
    cout << programming_language->print_line("Hello World") << endl;
    cout << programming_language->for_each("e", "vec") << endl
        << endl;
}
```

Gir utskriften:

```
cout << "Hello World" << endl;
for(auto &e: vec) {}
```

```
println!("Hello World");
for e in &vec {}
```

b) Utvid løsningen du skrev i oppgave a) slik at kildekoden:

```
vector<unique_ptr<ProgrammingLanguage>> programming_languages;
programming_languages.emplace_back(make_unique<Cpp>(make_unique<CMake>
()));
programming_languages.emplace_back(make_unique<Cpp>(make_unique<Meson>
()));
programming_languages.emplace_back(make_unique<Rust>
(make_unique<Cargo>()));

for (auto &programming_language : programming_languages) {
    cout << programming_language->build() << endl;
}
```

Gir utskriften:

```
cmake --build .
meson compile -C .
cargo build
```

**Skriv ditt svar her**

1	
---	--

--	--

---

Maks poeng: 10

### 3 Oppgave 3 (20%)

Lag klassen `Angle` slik at følgende kildekode:

```
cout << Angle<int>(30) << endl;  
cout << (Angle<int>(30) + 360) << endl;  
cout << (Angle<int>(30) + 3.14159265359) << endl;  
cout << Angle<double>(3.14159265359 / 2) << endl;  
cout << (Angle<double>(3.14159265359 / 2) + 3.14159265359) << endl;  
cout << (Angle<double>(3.14159265359 / 2) + 3) << endl;
```

Gir utskriften:

```
30  
30  
33  
1.5708  
1.5708  
1.4292
```

**Skriv ditt svar her**

1	
---	--

Maks poeng: 10

## 4 Oppgave 4 (20%)

Lag funksjonen `parallell_transform()` som returnerer en ny tabell avhengig av tabellene i første og andre argumentet, og funksjonen i tredje argumentet. Utrekningene skal skje i parallell. Du kan bruke `thread::hardware_concurrency()` for å finne antall CPU kjerner som er tilgjengelig. Følgende kildekode:

```
vector<int> a = {1, 2, 3, 4, 5};  
vector<int> b = {2, 3, 4, 5, 6};
```

```
cout << parallell_transform(a, b, [](int a_element, int b_element) {  
    return a_element + b_element;  
}) << endl;
```

```
cout << parallell_transform(a, b, [](int a_element, int b_element) {  
    return a_element * b_element;  
}) << endl;
```

Skal gi utskriften:

```
{ 3, 5, 7, 9, 11 }  
{ 2, 6, 12, 20, 30 }
```

**Skriv ditt svar her**

1	
---	--

Maks poeng: 10

