

Norges teknisk–naturvitenskapelige
universitet
Institutt for datateknologi og
informatikk

TDT4102 Prosedyre- og objektorientert programmering Vår 2022

Øving 0 for Windows

Frist: som for Øving 1

Mål for denne øvingen:

- Bli kjent med programmeringsverktøy
- Lage et første program med Visual Studio Code (VS Code)
- Kunne laste ned og kjøre eksempelprogram fra forelesningene med VS Code
- Lage et første program kun med teksteditor og kompilator
- Bli kjent med infobanken

Denne øvingen er en veiledning i å installere en programmeringsomgivelse slik at du kan skrive, redigere, compilere, debugge og kjøre et C++ program. **Det er nødvendig å gjennomføre og mestre det meste av det som gjennomgås i denne øvingen for å kunne utføre de obligatoriske øvingene.**

Vi vil våren 2022 benytte verktøyet VS Code som er gratis og kan brukes under Windows, MacOS og GNU/Linux.

Vær obs på at installering av alle programmene kan ta opptil flere timer, avhengig av hvor rask internettilkobling og PC du har.

Aktuelle kapitler i boka:

- Kapittel 0, 1 og 2 i Programming – Principles and Practice Using C++ (Second Edition)

0 Oppgave 0 - Installasjon

Hurtigoppsett

Dersom du er godt rutinert på installasjoner kan du følge den påfølgende forkortede installasjonsguiden. Vi vil likevel råde de aller fleste til å følge den detaljerte installasjonsguiden nedenfor.

1. Last ned VS Code for din plattform; <https://code.visualstudio.com/docs/?dv=win>

Viktig: Under installasjonen, sørg for å huke av for **Add to PATH**.

2. Gå til extension-menyen i VS Code (fire bokser i venstremenyen), og installer tillegget «TDT4102 Tools»
3. Godkjenn eventuelle pop-ups om tilganger og installer eventuelle programmer eller tjenester som tillegget ber om
4. Etter at installeringen av alle nødvendige programmer er fullført vil du få et spørsmål om å opprette en snarvei. Trykk *Yes*, og deretter *OK* etter at snarveien er opprettet.
5. Lukk VS Code og gå videre til oppgave 1 i denne øvingen

Detaljert installasjonsguide

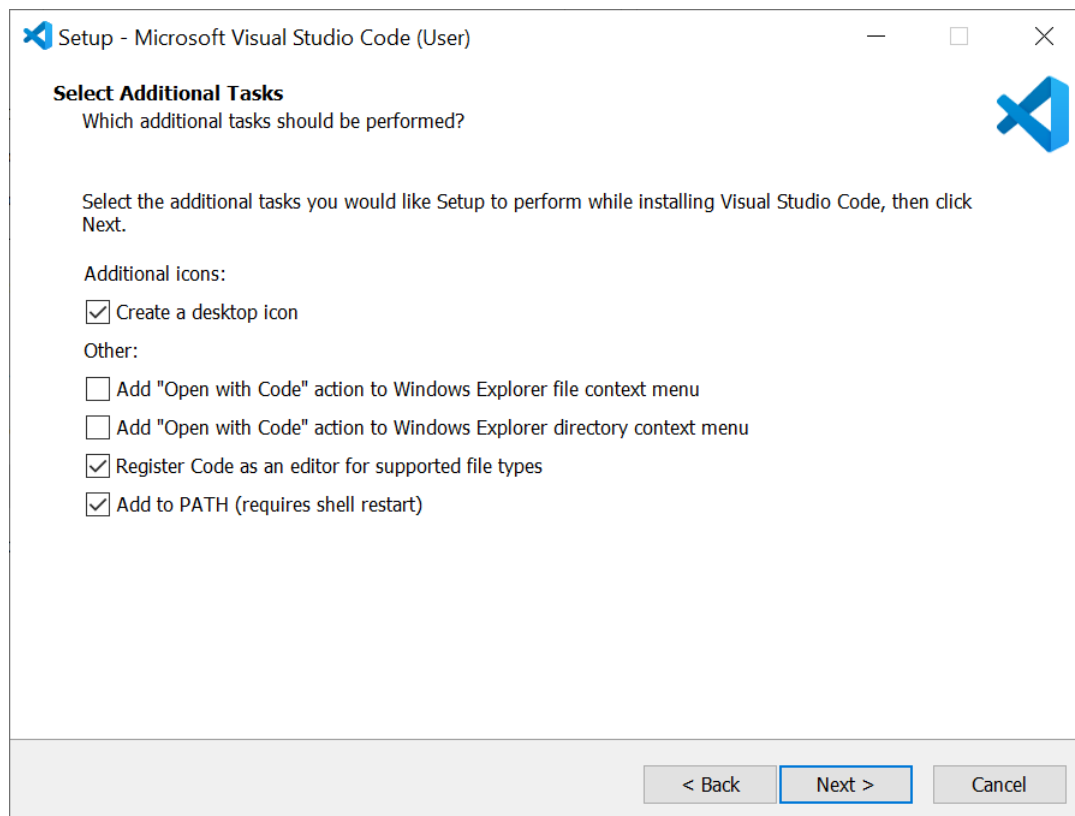
Les gjennom hele installasjonsinstruksen før du begynner.

Installasjonen krever at du installerer VS Code for å skrive og redigere kode. Oppstår det problemer underveis har vi listet opp noen kjente problemer i [Avsnitt 2](#).

VS Code er en teksteditor laget for å skrive og redigere kode i forskjellige programmeringsspråk. Programmet kan lastes ned og installeres fra <https://code.visualstudio.com/docs/?dv=win>. Enkelte nettlesere vil la deg kjøre installeringsprogrammet direkte, og andre vil lagre filen på PCen, sannsynligvis i mappen som heter *Nedlastinger* eller *Downloads*. Hvis installeringen av VS Code ikke starter automatisk etter nedlasting må du finne filen og dobbelklikke på den for å installere VS Code.

Under er en nummerert liste som tar for seg hvert steg av installasjonsprosessen. For å komme fra et punkt til det neste trykker man **Next** eller **Install**.

1. I første vindu av installeringen må du godta en lisensavtale fra Microsoft.
2. Videre får man valget om hvor du vil installere programmet. *Standardforslaget* er bra.
3. Så skal du velge hvor snarveiene skal ligge i Start-menyen. *Standardforslaget* er fint.



Figur 1: Marker minimum disse valgene for at VS Code skal oppføre seg som forventet. Merk at snarveien på skrivebordet ikke kan brukes for å bygge programmene i dette faget.

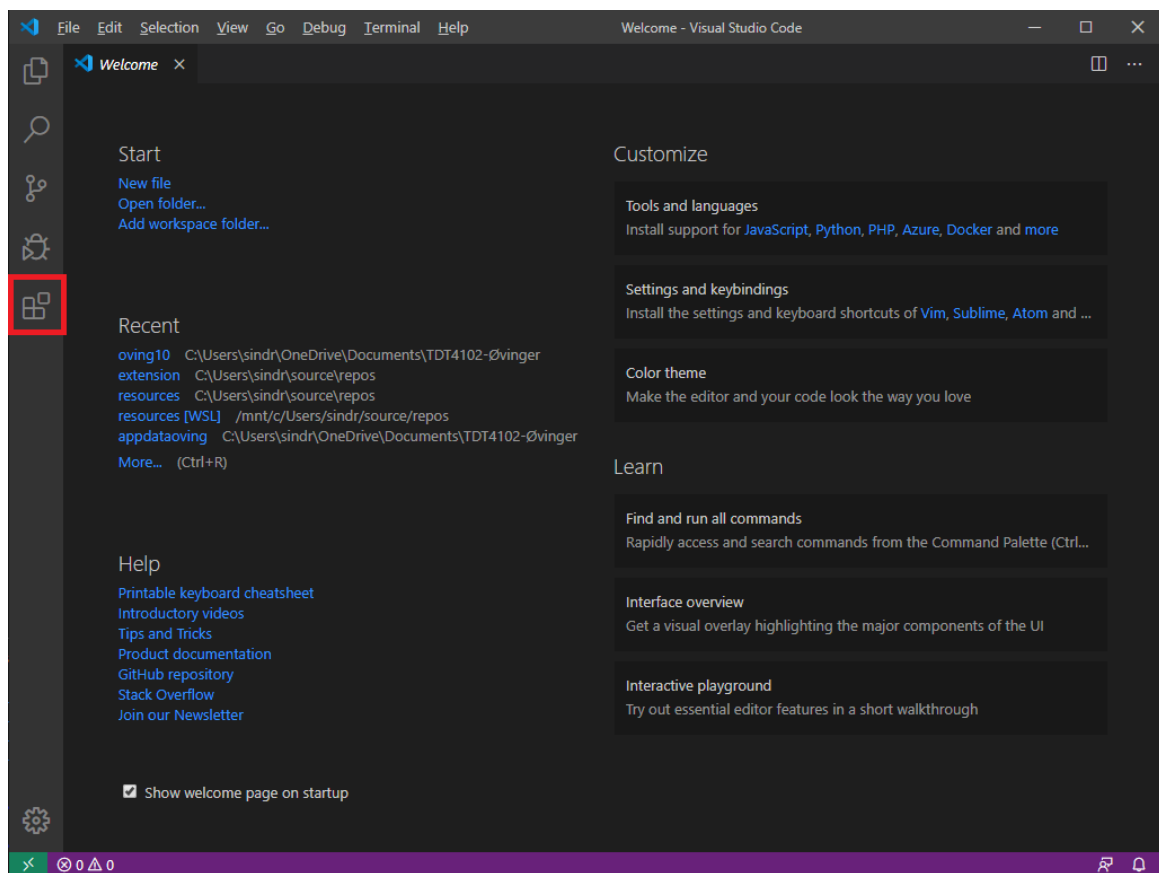
4. Nå skal du velge minimum de valgene som er markert i **Figur 1**.

Det viktigste at det er en hake på **Add to PATH** og det er en fordel å markere **Register code as an editor for supported file types**. **Add to PATH** lar code åpnes fra terminalen (dette er nødvendig for noe av oppsettet i faget). **Register code as an editor for supported file types** gjør at Windows bruker VS Code for å åpne utvalgte filtyper automatisk - for oss er det fint at kildekode i .cpp-filer og .h-/.hpp-filer åpnes automatisk i VS Code.

Merk at snarveien på skrivebordet ikke kan brukes til å åpne VS Code når du skal gjøre øvingene, mer om dette senere.

5. Når installeringen er fullført kan du trykke **Finish**.

VS Code skal nå starte og du er klar til å gå videre til neste steg.



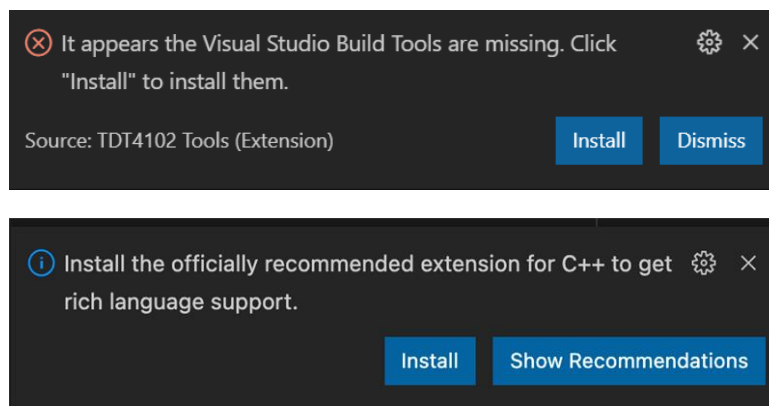
Figur 2: VS Code Extension-meny i rødt.

0.1 Installere TDT4102 Tools

VS Code vil la oss skrive og redigere kode, men vi trenger også en kompilator og bibliotek som følger med boken for å bygge og kjøre øvingene. For å forenkle denne prosessen har vi laget et tillegg (extension) til VS Code. Dette tillegget vil laste ned og starte installasjonen av de resterende verktøyene vi trenger, samt gjøre det lett å lage nye prosjekter. **Ikke lukk datamaskinen underveis i installasjonsprosessen, dette kan ta alt fra 10 minutter til et par timer.** *Det kan hende du må skrive inn passordet ditt flere ganger for at installasjonen skal kunne fullføre.*

Merk at stegene i denne prosessen kan variere hvis du allerede har installert VS Code, Xcode eller andre verktøy tidligere.

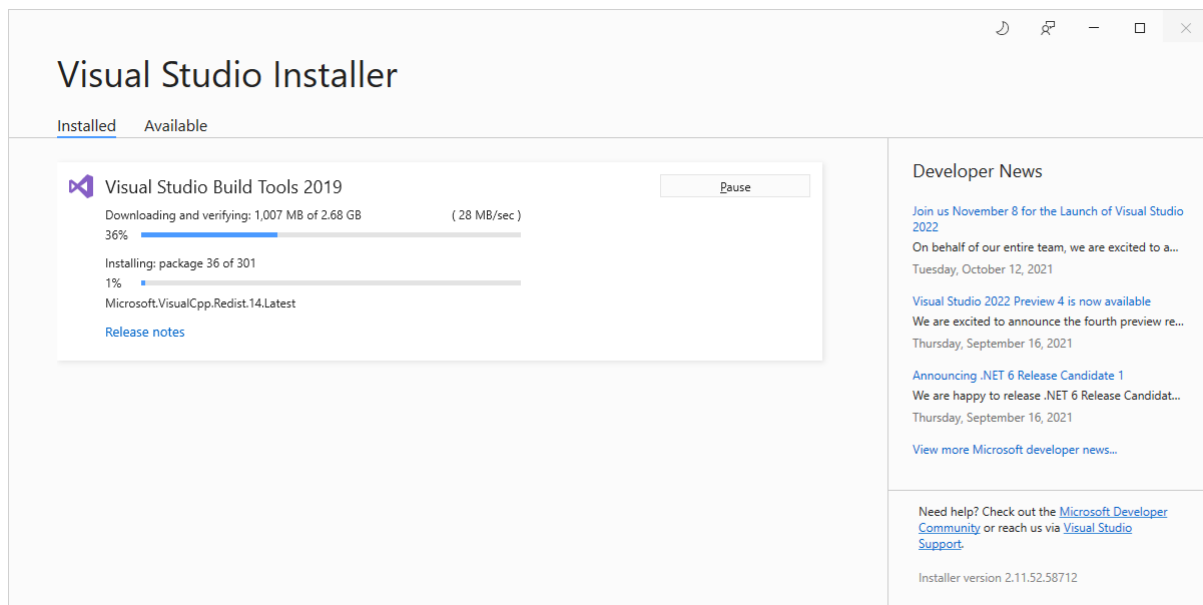
1. Åpne VS Code og velg Extensions fra menyen til venstre, se **Figur 2**. (Ctrl+Shift+X)
2. Søk etter TDT4102 Tools, og trykk på **Install**.
Nå vil alle nødvendige programmer lastes ned og installeres. Nederst til høyre i VS Code-vinduet vil det dukke opp en noen meldinger som forteller om installeringen er underveis, er fullført eller har feilet. I denne prosessen kan du bli spurt om du stoler på extensionen - her velger du **Trust Workplace & install**
3. I de fleste tilfeller får man en eller flere meldinger fra VS Code om at enkelte programmer eller tjenester mangler, som vist i **Figur 3**. Dersom dette skjer trykker du **Install** og deretter **Allow** eller **Install**, hvis du får spørsmål om å godkjenne tilgang. Under installasjonen vil du kunne se progresjonen i installasjonsvinduet, som i **Figur 4**



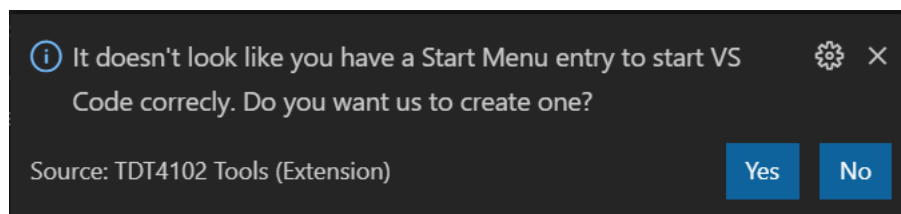
Figur 3: Manglende programmer under installasjon av extension i VS Code. Trykk **Install** og hvis du får spørsmål om å godkjenne tilgang, trykk **Allow**.

4. Etter at installeringen av alle nødvendige programmer er fullført vil du få et spørsmål om å opprette en snarvei, som i **Figur 5**. Trykk **Yes**.
5. Deretter kommer det opp en dialog som forteller at snarveien er opprettet. Trykk **OK**.
6. Lukk alle vinduer av VS Code før du går videre.

TDT4102-tillegget vil sjekke at du har de nyeste filene og malene, samt at alt du trenger er installert hver gang du åpner VS Code. Dersom det for eksempel skulle komme endringer i malene vil disse bli lastet ned automatisk, og du vil få en melding nederst i høyre hjørne av VS Code om at malene er oppdatert.



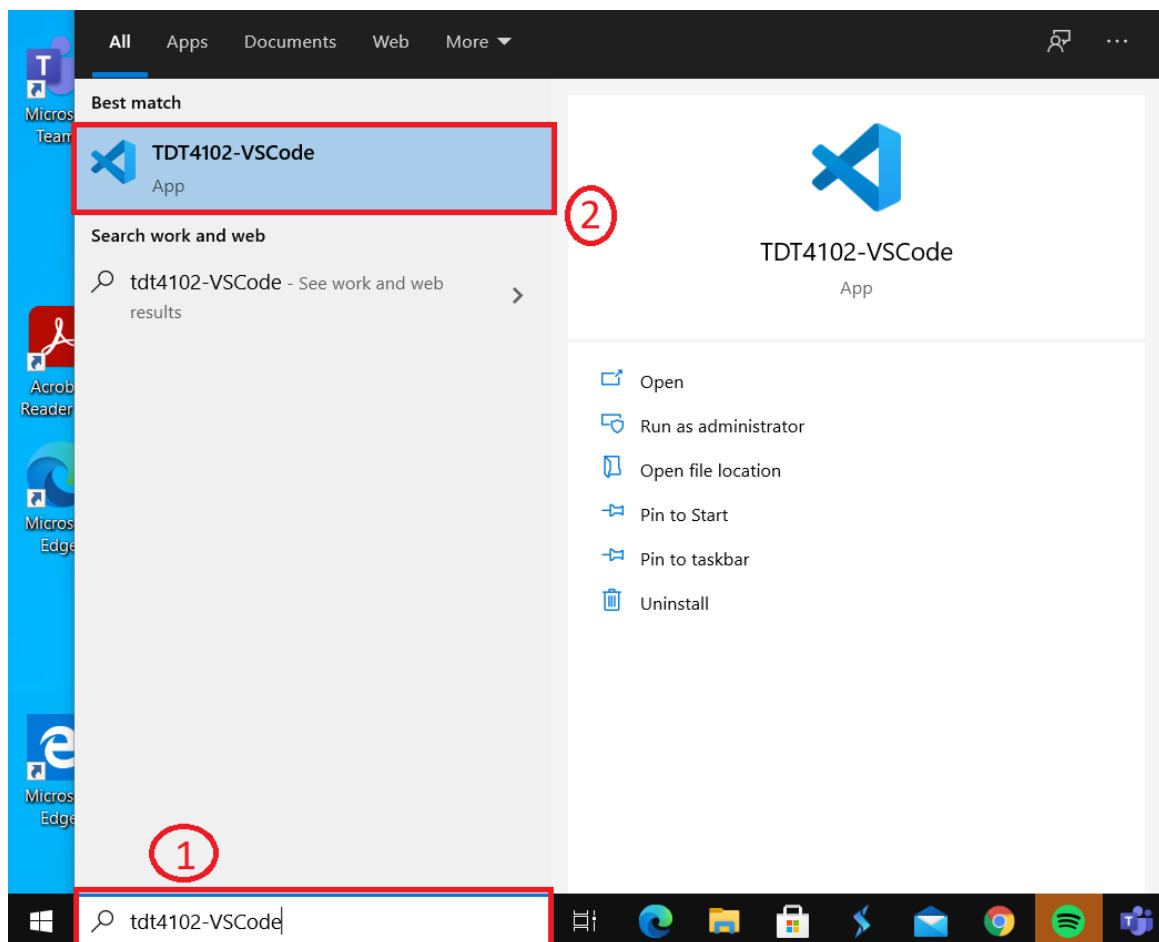
Figur 4: Installeringsvindu for VS Code.



Figur 5: Opprett en ny snarvei som åpner VS Code med alle nødvendige innstillinger. Trykk Yes.

Merk: Insider-version

VS Code kommer til å spørre om du vil bli med i insider-programmet. Takk **nei** til det.



Figur 6: Den nye snarveien lagt i Start-menyen etter installasjon av TDT4102 extension

1 Oppgave 1 - Programmering med VS Code

I denne oppgaven antas det at du allerede har installert VS Code, som beskrevet i Oppgave 0.

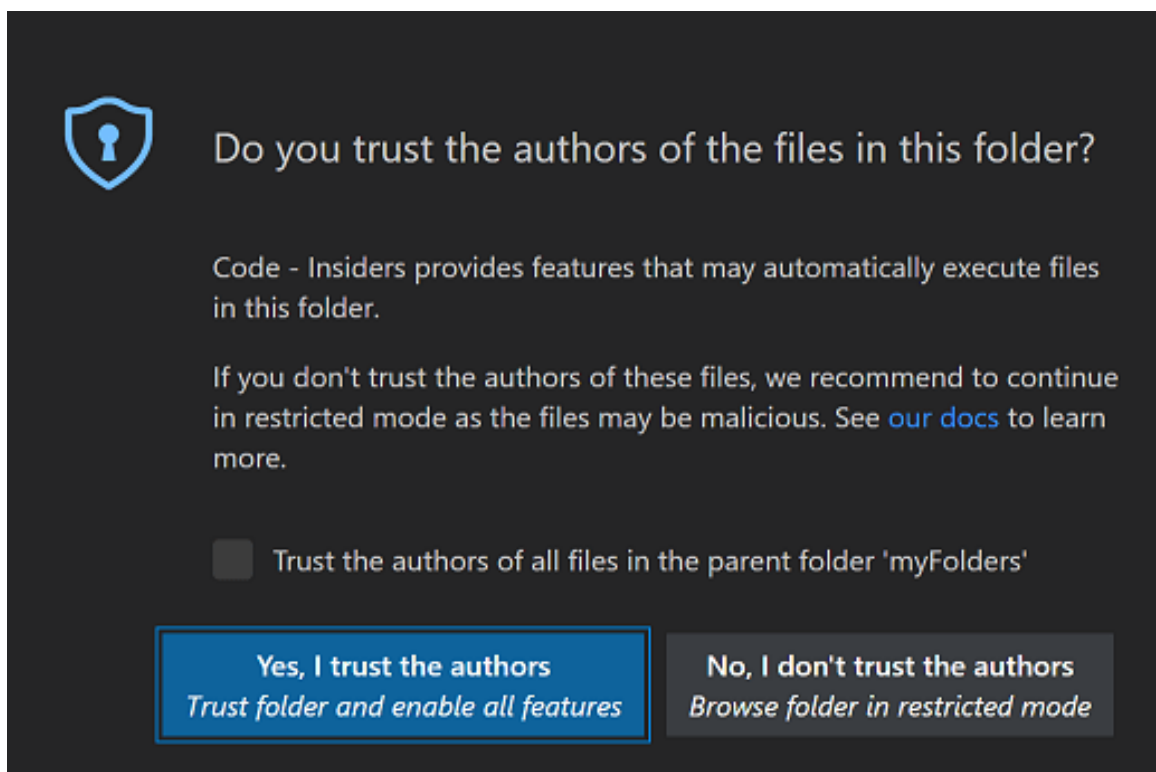
1.1 Oppgave 1.1 - Mitt første prosjekt i VS Code

1. Start VS Code fra snarveien, TDT4102-VSCode, som ligger i Start-menyen, som vist i **Figur 6**. Åpne Start-menyen og begynn å skrive `tdt4102`, (1) i figuren, da skal snarveien ligge på toppen av menyen, (2) i figuren.

*For å lettere finne tilbake til snarveien kan du feste den til oppgavelinjen: høyreklikk på snarveien, velg **Mer** og **Fest til oppgavelinjen**.*

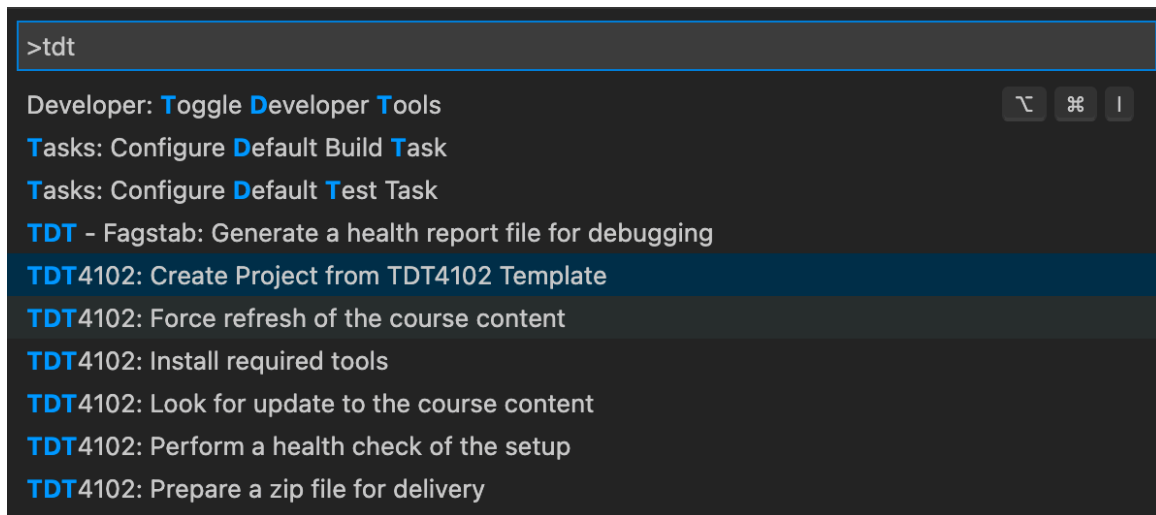
2. Når du har trykket på snarveien og VS Code starter, vil det også åpnes et konsollvindu - dette kan lukkes etter at VS Code har startet opp.
3. Lag en ny tom mappe et valgfritt sted (f.eks. i Documents-mappen din). Gå inn i VS Code og trykk på **File** øverst til venstre, deretter trykk **Open Folder**. Velg mappen du laget.

Hvis du blir spurt om du stoler på mappen **7**, trykk på **Yes, I trust the authors**. For mer info klikk [her](#).



Figur 7: Meny hvor du kan gi tillit til arbeidsmappen.

4. Trykk **Ctrl+Shift+P** for å få opp *kommandoboksen*. Begynn å skrive `TDT4102: Create Project from TDT4102 Template` og trykk **Enter** når du har markert valget. Velg deretter **Blank project**. Se **Figur 8** for hvordan kommandoboksen ser ut etter å ha skrevet `tdt`.



Figur 8: Kommandoboksen (Command Palette) etter å ha skrevet `tdt` første gang.

5. Til venstre i VS Code vil du nå se en oversikt over filene i prosjektmappen din. Her skal det ligge tre elementer: filene `main.cpp` og `Makefile`, og mappen `.vscode`.

Når tillegget gjenkjenner et korrekt oppsatt prosjekt vil det stå **TDT4102 Project** ✓ nederst i venstre hjørne av VS Code. For å sjekke at du har åpnet VS Code riktig og at alt er som det skal kan du alltid se etter denne teksten og haken når du har åpnet et prosjekt som er korrekt oppsatt for øvingsopplegget. Det vil for eksempel ikke være synlig i en tom mappe.

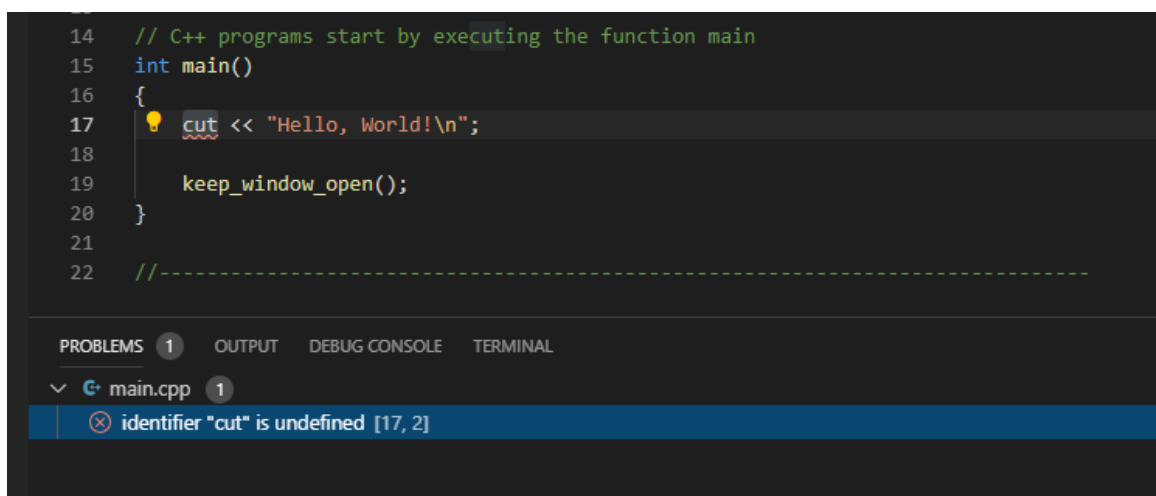
6. Filen `Makefile` og mappen `.vscode` inneholder innstillinger som bestemmer hvordan VS Code skal compilere og kjøre koden i prosjektet ditt. Vi anbefaler at du ikke endrer disse filene. Hvis du mot formodning skulle endre noen av disse filene kan du skrive `Ctrl+Shift+P`, skrive **TDT4102: Create Project from TDT4102 Template** og velg **Configuration only**. Alternativt kan du gjøre **Punkt 4** om igjen og velge **Overwrite** for alle spørsmålene du får, bortsett fra når du får spørsmål om å skrive over `main.cpp`.
7. Det viktigste for oss er `main.cpp`, i denne ligger kildekoden til et enkelt program som skriver "Hello, World!" til skjermen. Hvis du trykker på `main.cpp` vil filen åpnes som en fane i VS Code så du kan se og redigere den.
8. Kjør programmet ditt ved å trykke `Ctrl+F5` (det kan være du må trykke **fn**-tasten for å få tilgang til F-tastene, det blir i så fall **fn+Ctrl+F5**), eller ved å gå inn i menyen **Run** og velge **Run Without Debugging**. I terminalvinduet på bunnen (den kan flyttes) av VS Code vil teksten "Hello, World!" være skrevet. Hvis du ikke kan se terminalen trykk på Terminal i menyen øverst, trykk så New Terminal og kjør programmet igjen. (Det opprettes mapper i prosjektet som heter **Debug** og **Release** når programmene bygges. Disse mappene inneholder det byggede programmet for hhv. debug-versjon og release-versjon av programmet. Enn så lenge kan disse mappene ignoreres.)

1.2 Oppgave 1.2 - Kompilering og feilmeldinger

En god del av tiden du bruker på å gjøre øvinger vil bestå i å finne ut hva som er feil i koden din. En type feil er syntaktiske feil (enkle skrivefeil) som resulterer i kode som ikke vil kompilere. Hvis du prøver å kompilere kode som ikke er riktig skrevet vil kompilatoren gi deg feilmelding(er) som inneholder informasjon om hva som kan være galt. Noen ganger er dette forståelig informasjon, andre ganger kan det være vanskelig å skjønne feilmeldingene.

Du skal nå med vilje «ødelegge» koden din ved å endre på småting, og deretter observere hva slags feilmeldinger du får når du kompilerer. Du kan for eksempel gjøre følgende:

1. Slett semikolonet på slutten av linjen som skriver ut teksten «Hello World!» og kompiler på nytt.
2. Det vil nå dukke opp en feilmelding i terminalvinduet nederst, forstår du feilmeldingen?
3. Introduser andre feil og les feilmeldingene som kompilatoren gir. Du kan f.eks. prøve å slette en av krøllparentesene, skrive `cout` som `cut` osv.
4. Det skal også dukke opp feilmeldinger i Problems-fanen nederst i VS Code, se [Figur 9](#). Her kan du klikke på linjer i feilmeldingen for å se hvordan VS Code markerer linjene i koden din der den tror feilen ligger.
5. Husk å rette opp feilene i filen igjen før du går videre.



Figur 9: Problems-fanen i VSCode

NB: Det er ikke alltid Problems-fanen er pålitelig

Problems-fanen er fin for å fremheve syntaksfeil i koden din, når den fungerer. Det finnes dessverre tilfeller der fanen enten ikke fanger opp alle feil, eller at den til og med uthever ikke-feil i koden din.

Fasiten får du alltid ved kompilering av koden din, som skjer når du *bygger* prosjektet med **Ctrl+Shift+B**, eller ved kjøring av programmet. Alle feil du eventuelt har i koden din vises da som feilmeldinger i *terminalen* - denne er alltid pålitelig. Ta derfor Problems-fanen alltid med en klype salt.

1.3 Oppgave 1.3 - Eksempelprogrammer fra forelesningene

Ved å åpne kommandovienduet (**Ctrl+Shift+P**) og skrive **TDT4102: Create Project from TDT4102 Template** vil du få opp valg om å opprette et nytt prosjekt. Til nå har vi brukt malen **Blank project**. Det er også to valg som heter **Examples** og **Configuration only**. Ved å velge **Examples** vil det dukke opp en liste med eksempelprogrammene fra forelesningene. Disse kan åpnes på samme måte som andre maler. Eksemplene inneholder ikke prosjektinnstillinger som **.vscode**-mappen eller **Makefile**. Disse prosjektinnstillingene finnes i malen **Configuration only** og den bruker vi for å gjøre eksempelprogrammene komplette slik at de kan kjøres fra VS Code.

For å åpne og kjøre et eksempel vil vi:

1. Lage en ny mappe og åpne den i VS code som beskrevet i oppgave 1.1.3
2. Opprette et prosjekt med malen **Configuration only** (**.vscode** og **Makefile**)
 - **Ctrl+Shift+P** og velg **TDT4102 Create Project from TDT4102 Template**
 - Velg **Configuration Only**
3. Åpne en av malene i **Eksempler** mappen.
 - **Ctrl+Shift+P** og velg **TDT4102 Create Project from TDT4102 Template**
 - Velg **Examples**
 - Skriv inn **hello_graphics** i søkefeltet og velg eksempelet
4. Kjør eksempelprogrammet ved å trykke **Ctrl+F5** (det kan være du må trykke **fn**-tasten for å få tilgang til F-tastene, det blir i så fall **fn+Ctrl+F5**), eller ved å gå inn i menyen **Run** og velge **Run Without Debugging**.

Hvis alt har gått som det skal vil det nå åpnes et nytt vindu med en blå sirkel. Dette er et eksempel på enkel grafikk, som vi kommer til å bruke en del senere i øvingsopplegget.

1.4 Oppgave 1.4 - Lage .zip for innlevering

Tillegget TDT4102 Tools har en funksjon for å pakke filene i øvingen til en **.zip**-fil som kan leveres på BlackBoard. Trykk på **Ctrl+Shift+P** og skriv **TDT4102: Prepare a zip file for delivery**. Trykk enter. En **.zip**-fil med navn **Handin.zip** vil dukke opp i margin til venstre og ligge i mappen med øvingen. Du må selv laste opp filen på BlackBoard for å få godkjent øvingen.

NB! Du skal ikke levere noe for denne øvingen, men test ut at det fungerer å lage en **.zip**-fil.

1.5 Oppgave 1.5 - Infobanken

En av funksjonene til TDT4102 Tools gir det tilgang til fagets infobank. Her vil du kunne få tilgang på nyttige ressurser som artikler, videoer og øvinger. Formålet er at du enkelt skal kunne finne det du trenger inne i **VS Code**.

For å åpne infobanken trykk på **Ctrl+Shift+P** og skriv **TDT4102: Open infobank**. En liste over ulike ressurser vil dukke opp hvor du kan søke eller bla/scrolle til den ressursen du ønsker å bruke. Du kan søke både etter tittel eller nøkkelord.

Prøv å åpne følgende ressurser:

1. **Piazza** - fagets forum
2. **Graph_lib** dokumentasjon
3. **Feilmeldinger** - en artikkel som hjelper med debugging
4. **Hello graphics** - her finner du lenke til en video som forklarer hvordan man lager grafikk i c++
5. **Øving 0** - Du kan til og med finne denne øvingen i infobanken! Øvingene i faget vil bli lagt ut i infobanken i løpet av året

1.6 Oppgave 1.6 - Kompilering fra kommandolinje (frivillig oppgave)

Denne oppgaven er ikke nødvendig for å følge øvingsopplegget videre, men gir bedre innsikt i hvordan kompilering og kjøring av C++ kode fungerer, uten at Visual Studio Code gjør det for deg.

Skriving av kildekode og kompilering kan i prinsippet gjøres med enkle verktøy. En helt vanlig teksteditor er alt du trenger for å skrive kode, og kompilering kan gjøres ved å kjøre kompilatoren fra kommandolinja. Vi velger å ikke benytte *Graph.h* eller *Simple_windows.h* i dette kommandolinje-eksempelet, ettersom det er mer avansert å kompilere "for hånd". Eksempelprogrammet vi skal bruke i denne oppgaven vises i **Figur 10**.

```
#include <iostream>
int main() {
    std::cout << "Hello World" << std::endl;
}
```

Figur 10: Eksempelprogram uten avhengigheter på Graph.h eller Simple_window.h

Vi vil her bruke Microsofts kompilator **cl**, som fulgte med under installasjonsprosessen.

1. Opprett en ny mappe i hjemmemappen din, og åpne denne i Visual Studio Code. Kopier og lim inn Hello World-eksemplet i Figur 10 i en ny fil, for eksempel med navnet **HelloWorld.cpp**. Sjekk mappen hvor filen din er lagret, og forsikre deg om at den er der og har riktig navn.

Det er vanlig konvensjon at kildekodefiler for C++ har filtypenavnet (delen av filnavnet etter punktum) `cpp`. Ved å bruke riktig filtypenavn oppnår du at mange verktøy automatisk skjønner at filen inneholder C++-kode. Hvis du bruker et annet filtypenavn kan det hende at du ikke får kompilert koden.

2. For å kompilere fra kommandolinjen trenger vi først å starte opp et kommandolinjevindu. Dersom du åpner kommandolinjen på vanlig måte med `cmd.exe`, vil man hver gang man ønsker å kjøre C++-kompilatoren `cl` måtte spesifisere hvilken mappe denne ligger i. Det er tungvint. I stedet skal vi bruke «Developer Command Prompt for VS2019». For å starte denne, gå inn på start-menyen og begynn å skrive «Developer Command Prompt». Du skal få opp programmet i listen – start dette.
3. Vi skal nå navigere oss fram til mappen der du lagret `HelloWorld.cpp`. For å navigere mellom mapper i Windows-kommandolinjen bruker vi disse kommandoene
 - **cd** (change directory)
 - For å gå inn i en mappe skriver man: `cd <navn på mappen din>`
 - for å gå opp et nivå skriver man: `cd..`
 - **dir**
 - for å se hva som ligger i mappen man er i skriver man `dir`

Når du først starter «Developer Command Prompt» vil du antakelig befinne deg i mappen som Visual Studio 2019 er installert i, for eksempel `C:\Program Files\Visual Studio 2019`. Dersom mappen du lagret `HelloWorld.cpp` i ligger på en annen disk enn Visual Studio, må du først bytte til den. Det gjør du ved å skrive stasjonsbokstaven etterfulgt av kolon, f.eks. `cd D:\`. Hvis den ligger på samme disk, kan du nå gå «nedover» i mappehierarkiet ved å skrive kommandoen `cd ..` (to punktum) helt til du havner i *roten* av disken, f.eks. `C:\`.

Gå nå «oppover» i mappehierarkiet til du kommer til mappen der du lagret `HelloWorld.cpp`, ved å igjen bruke `cd`, nå med navnet på mappen du vil gå inn i. Hvis `HelloWorld.cpp` ligger inni hjemmemappen din, vil du for eksempel først skrive `cd Users`, så `cd brukernavn`, og så videre. For å gå inn i mapper med mellomrom i navnet, kan du bruke hermetegn rundt mappenavnet. For å se hvilke filer og mapper som ligger i mappen du befinner deg i, kan du bruke kommandoen `dir`.

4. Når du er kommet til mappen der `HelloWorld.cpp` ligger, skal vi kompilere denne til en programfil. For å gjøre dette skriver du

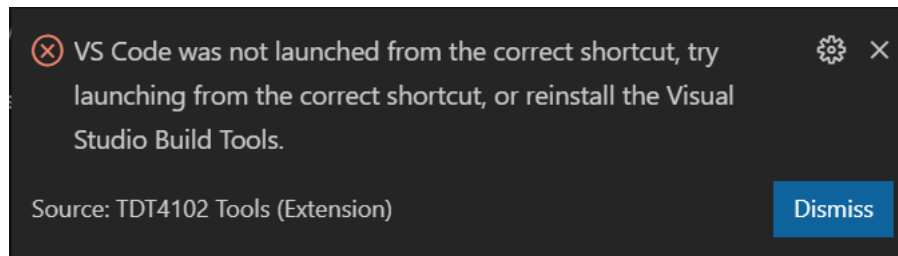
```
cl HelloWorld.cpp
```

5. Sjekk nå hva som ligger i mappen og se hvilke filer som er produsert. `obj`-filen er en midlertidig fil du ikke trenger å bry deg om, mens `exe`-filen er det endelige programmet ditt. Kjør programmet du har laget ved å skrive `HelloWorld.exe`, og sjekk utskriften.

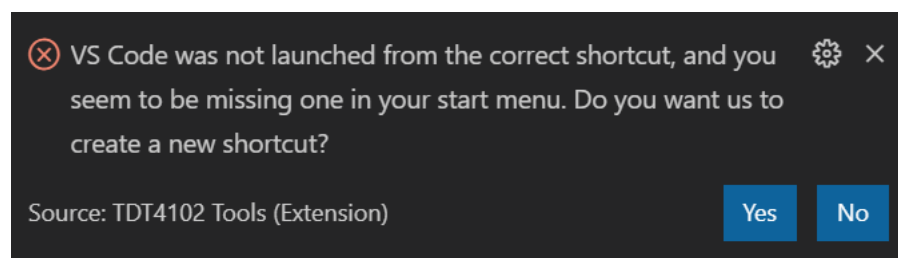
Her kompilerte vi et veldig enkelt program som besto av bare en fil, og helt uten grafikk. Man kan i prinsippet gjøre det samme for større programmer, men det krever at man spesifiserer hvordan filene skal lenkes sammen, hvor grafikkbiblioteket ligger og noen andre detaljer som er tungvint å gjøre for hånd. Makefilen som ligger i det utdelte prosjektet gjør at alt dette skjer automatisk så man slipper å tenke på det.

2 Feil og andre meldinger under installering

2.1 VS Code was not launched from the correct shortcut...



Figur 11: VS Code er ikke åpnet fra snarveien i Figur 6.



Figur 12: Snarveien mangler og må installeres.

Får du en feilmelding som sier at VS Code ikke ble startet fra riktig snarvei? Hvis den eksisterer, men VS Code ble startet på annet vis får du meldingen i Figur 11. Hvis snarveien ikke er installert vil du få meldingen i Figur 12.

2.1.1 Snarveien er installert - Hvordan åpne fra den utdelte snarveien?

1. Lukk alle vinduer av VS Code.
2. Følg anvisningene i Figur 6.

2.1.2 Snarveien er ikke installert - Installer snarveien

Hvis du ikke finner snarveien eller den ikke fungerer må snarveien installerer på nytt.

1. Lukk alle vinduer av VS Code.
2. Følg anvisningene i Avsnitt 0.1.

2.2 Bruk av skylagring og antivirus

Hvis du dukker opp i problemer, kan dette være på grunn av bruk av skylagringstjenester (iCloud, OneDrive, Dropbox, etc.) Sørg for at skylagringstjenestene ikke virker på arbeidsfilene.

3 Nyttig å vite

3.1 Auto save

Vi anbefaler sterkt å bruke auto save i dette faget. For å skru det av og på gå til menyen øverst og trykk på File så på **Auto save**. Hvis Auto Save er av vil TDT Tools minne deg om å skru det på.