**Årsplan i *programmering valgfag* ved Byremo ungdomsskole**

Årstall: 2022/2023

Læreverk:

Faglærer: Petter Valderhaug

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Verdigrunnlag:** | | |
| **Kompetansemål:** Gjøre rede for hvordan datamaskiner og dataprogram fungerer, og forklare egne og andres koder | | |
| **Kjerneelement:** | | |
|  | **Læringsmål, Tverrfaglige tema og innhold i opplæringen;** hva skal det arbeides med | **Arbeidsmåter –** hvordan skal en arbeide for å oppnå målene. (Praktisk, skriftlig, konkretiseringer, samspill) | **Vurdering** |
| **Uke**  **34 - 41** | Introduksjon til programmering.     * bruke flere programmeringsspråk der minst ett er tekstbasert      * bruke grunnleggende prinsipper i programmering, slik som løkker, tester, variabler, funksjoner og enkel brukerinteraksjon | Introduksjon til programmerings-faget og hva programmering vil si.  Presentere våre forhold til programmerte ting  Hvordan skal vi jobbe med koding i løpet av dette året?  Få praktisk kjennskap til flere sentrale begreper innen koding  ▪ Variabler  ▪ Lister  ▪ Datatyper  ▪ Input  ▪ If/else-setning  ▪ Operasjon  ▪ Løkker og lister  ▪ Funksjoner og funksjonalitet  ▪ By-løkker    Vi får praktisk erfaring om disse begrepene gjennom bruk av dem Scratch. Elevene har langsiktig prosjekt hvor de skal produsere en lengre animasjon eller et spill. Dette arbeidet skal jobbes med kontinuerlig og skal gis tilbakemeldinger og forbedringspotensialet. Arbeidet skal besøkes senere i året – elevene får mulighet til å forbedre det gjennom de kunnskaper de har skaffet i løpet av året. | Individuell måloppnåelse på spill/animasjon på Scratch |
|  | **Verdigrunnlag:** | | |
| **Kompetansemål:** Bruke flere programmeringsspråk, deriblant minst ett som er tekstbasert.    Bruke grunnleggende prinsipper i programmering, slik som variabler, løkker, vilkår og funksjoner, og reflektere over bruken av disse | | |
| **Kjerneelement:** | | |
|  | **Læringsmål og innhold i opplæringen;** hva skal det arbeides med | **Arbeidsmåter –** hvordan skal en arbeide for å oppnå målene. (Praktisk, skriftlig, konkretiseringer, samspill) | **Vurdering** |
| **Uke**  **42-50**  **(7 økter)** | * utvikle og feilsøke programmer som løser definerte problemer, inkludert realfaglige problemstillinger og kontrollering eller simulering av fysiske objekter * Tolkning av kode uten en praktisk visning av hva den gjør. | Jobbe med Python. Få kjennskap til Python – hvordan python fungerer og hvordan vi skal jobbe med python fremover    Bruke Codecombat.com og BBC’s micro:bit for praktisk visning av den koden elevene skriver.  Jobbe videre med tolkning og forståelse av koder  Forsøke å bruke MU Editor for å skrive frihånds python kode  Kunne å skrive et program i MU editor ved hjelp av guides og google søk.    Valgfagsdag uke 45  Vise elevene en kode skapt i Scratch – utfordre elevene til å tolke hva koden blir å gjøre uten at de ser hva den gjør. Etter elevenes innspill spille av koden for å se om elevene har tolket den riktig  <https://automatetheboringstuff.com/#toc>  god bok for jobbing med Python.  Uke 50 Juleverksted: Elevene skal ved hjelp av scratch animere en scene fra «A Christmas Carol» fra Charles Dickens | Gloseprøve på kjernebegrep innen koding |
|  | **Verdigrunnlag:** | | |
| **Kompetansemål:** Analysere problemer, gjøre dem om til delproblemer og gjøre rede for hvordan noen av delproblemene kan løses med programmering | | |
| **Kjerneelement:** | | |
|  | **Læringsmål og innhold i opplæringen;** hva skal det arbeides med | **Arbeidsmåter –** hvordan skal en arbeide for å oppnå målene. (Praktisk, skriftlig, konkretiseringer, samspill) | **Vurdering** |
| **Uke**  **1-10** | * bruke flere programmeringsspråk der minst ett er tekstbasert * utvikle og feilsøke programmer som løser definerte problemer, inkludert realfaglige problemstillinger og kontrollering eller simulering av fysiske objekter * omgjøre problemer til konkrete delproblemer, vurdere hvilke delproblemer som lar seg løse digitalt, og utforme løsninger for disse | Videreutvikle forståelsen av tekstbasert kode. Kunne bruke Micro:bit sitt online redskap for å skape en hinderløype for Micro:bit bil å kjøre på.    Forsøke å bruke Python for å automatisere en handling – organisere filer på dataen til mapper (Alle tekst format i en, alle bilder i en annen og alle lydfiler i en tredje).    Gå gjennom Minecraft Education – hvordan kunne bruke MC Edu for å lære om koding og videreutvikle kode, få kjennskap til ofte brukte komandoer innen python som Input(), list(), len(), print() | Vurdering: Bruke det vi har lært av python til å skape et program som gjør en bevist handling |
|  | **Verdigrunnlag:** | | |
| **Kompetansemål:** Planlegge og skape et digitalt produkt og vurdere dette med tanke på brukervennlighet    Utvikle og feilsøke dataprogram som løser definerte problemer, inkludert kontrollering eller simulering av fysiske objekter | | |
| **Kjerneelement:** | | |
|  | **Læringsmål og innhold i opplæringen;** hva skal det arbeides med | **Arbeidsmåter –** hvordan skal en arbeide for å oppnå målene. (Praktisk, skriftlig, konkretiseringer, samspill) | **Vurdering** |
| **Uke**  **10-24** | ▪ gjøre rede for hvordan datamaskiner og programmer fungerer, inkludert et utvalg utbredte programmeringsspråk og deres bruksområder    ▪ dokumentere og forklare programkode gjennom å skrive hensiktsmessige kommentarer og ved å presentere egen og andres kode | Terpe på alle formene for koding vi har jobba på iløpet av året: Scratch, MU, Micro:bit, Minecraft Education    Elevene får mulighet til å videreutvikle spillene eller animasjonene de lagde i starten av året – nå med den nye kunnskapen de har om koding og mulighetene.    Om noen får det til, prøve å lage et spill via Python – tekst basert istedenfor blokk basert.    Vært gjennom alle pakkene på Minecraft Education.    Mulighet for å forbedre tidligere prosjekter    Se på brukervennlighet av forskjellige apper. | Generell arbeidsinnsats  Mulighet for forbedring av tidligere arbeid. |