

Obligatorisk oppgave nr.4

PROG1001 - Grunnleggende programmering, høsten 2024

Innleveringsfrist: 14.november 2024 kl.11:00 (må overholdes)
i Blackboard på pdf-format

Du *skal* bruke/ta utgangspunkt i `oblig4.tpl`.

Bruk gjerne også `LesData.h` som introduseres i `EKS_34.c`.

Oppgaven går ut på å registrere lag, kamper og resultater i et seriespill (f.eks. fotball). Brukeren skal kunne legge til nye lag/kamper, skrive ut lag/kamper og registrere/slette resultatet for en kamp. (Det skal *ikke* lages en serietabell ut fra resultatene.)

Av vedlegget kan man se at det skal lages innmat til følgende 16 funksjoner:

`void nyttLag()` og `void lagLesData(struct Lag* lag, char* nv)`

Den første funksjonen kommer med en melding om det ikke er plass til flere lag i datastrukturen. Den leser så inn navnet på det nye laget. Hvis det allerede finnes et lag med det gitte navnet får brukeren beskjed og funksjonen avslutter. I motsatt fall opprettes et nytt lag og antall lag telles opp med en. Den andre funksjonen setter lagets navn til å være parameteren, og nullstiller alle de andre datamedlemmene.

`void nyKamp()` og `void kampLesData(struct Kamp* kamp)`

Den første funksjonen kommer med en melding om det ikke er plass til flere kamper i datastrukturen. Den kommer også med en egen melding om det ikke er minst to lag allerede registrert. I motsatt fall opprettes en ny kamp, den leser selv inn alle sine data vha. den andre funksjonen, og antall kamper telles opp. Den andre funksjonen sørger for at det blir satt pekere til to lovlig navngitte lag, og at resten av datamedlemmene nullstilles/false.

**`void skrivKamper()`, `void kampSkrivData(const struct Kamp* kamp)`,
`void lagSkrivNavn(const struct Lag* lag)` og
`bool kampFerdig(const struct Kamp* kamp)`**

Den første funksjonen kommer med en melding om ingen kamper er registrert. I motsatt fall skrives *alle data* ut (vha. den andre og tredje funksjonen) om *alle kampene* som er *ferdig spilt* (den fjerde funksjonen gir svar på dette).

void skrivAlleLag() og **void lagSkrivData(const struct Lag* lag)**

Den første funksjonen kommer med en melding om ingen lag er registrert. I motsatt fall skrives *alle data* om *alle lagene* ut på skjermen, bl.a. ved å bruke den andre funksjonen.

void registrerResultat(),

void kampRegistrerResultat(struct Kamp* kamp) og

void lagGiPoeng(struct Lag* lag, const int scoret, const int inn)

Den første funksjonen kommer med en melding om ingen kamper finnes. I motsatt fall leses et kampnummer, inkl. 0 (null). Skrives et reelt kampnummer (ulikt 0), så leses resultatet inn for denne kampen vha. den andre funksjonen. Denne funksjonen sjekker først om kampen allerede er ferdig/spilt. Om så er tilfelle, kommer det bare en melding. I motsatt fall skrives begge lagenes navn (vha. tidligere laget funksjon) og deres antall mål scoret leses inn. Så kaller den andre funksjonen den tredje og ganger, med litt ulike parametre, slik at begge lagene tildeles 0, 1 eller 3 poeng ut fra kampresultatet. Kampen registreres som «ferdig», og alle dens nåværende data skrives til slutt ut på skjermen (vha. tidligere laget funksjon).

void slettResultat(),

void kampSlettResultat(struct Kamp* kamp) og

void lagFjernPoeng(struct Lag lag, const int scoret, const int inn)

Den første funksjonen gjør det samme som `registrerResultat()`, bare at `kampSlettResultat(...)` kalles i stedet. Denne funksjonen kommer bare med en melding om kampen ikke er ferdig/spilt. I motsatt fall fratas lagene de registrerte poengene (ut fra det hittil registrerte resultatet, og ved at den tredje funksjonen kalles to ganger), kampen settes til å ikke være ferdig/spilt, samt at kampens antall mål nullstilles.

Lykke til!

Karianne Kjørås, Sondre Gunneng & FrodeH