

1. Napisati potprogram u asemblerском језику којим се računa n-ти елемент Fibonačijevog низа:
  - a. Заглавље функције (потпрограма) дато је у главни.c фајлу у Z1 folderу
  - b. Потпрограм као повратну вредност враћа код грешке – грешка наступа уколико дође до прекопрачења при сабирању (за веžбу, можете проширити задатак тако да грешка наступа и када је прослеђено невалидно n); потпрограм враћа 0 уколико није наступила грешка, а било шта друго уколико јесте наступила
  - c. Потпрограм враћа пронађени n-ти елемент низа преко параметра *unsigned int\** *rez* пренетог по адреси
  - d. Фајл *resenje\_1.S* садржи решење са мање укључених параметара, док фајл *resenje\_2\_za\_testiranje.S* садржи решење које се може проследити скрипти за тестирање
2. Написати потпрограм у аsemblerском језику којим се računa n-ти елемент Fibonačijevog низа коришћењем **rekurzivnog алгоритма**. (Детаљније о рекурзивном Fibonačijevom алгоритму можете наћи на sledećем linkу:

<https://medium.com/launch-school/recursive-fibonacci-method-explained-d82215c5498e>

Osnovна идеја јесте да се n-ти елемент низа računa по формулі:

$$F(n) = F(n-1) + F(n-2), \text{ за } n > 2 \text{ уколико се низ индексира почијев од 1, а } F(1) = F(2) = 1$$

- a. Заглавље функције (потпрограма) дато је у главни.c фајлу у Z2 folderу
- b. Потпрограм резултате враћа обрнуто од задатка 1 – пронађени n-ти елемент враћа се као повратна вредност потпрограма, док се грешка враћа преко параметра пренетог по адреси
- c. **Pomoć** – постоје три важне целине у овом потпрограму:
  - i. Постављање услова изласка из рекурзије – треба га поставити на врху, како би се рекурзија на време прекинула; услов се огледа у провери да ли је trenutno n > 2, и да ли треба улазити у даљу рекурзију
  - ii. Позив потпрограма за F(n-1)
  - iii. Позив потпрограма за F(n-2) и сабирање резултата са резултатом позива F(n-1)
- d. **Nапомена:** решење доступно на ACS-у не садржи код који узима у обзир грешку, па га треба проширити како би одговарао прiloženom .c фајлу
3. Написати потпрограм у аsemblerу којим се покомпонентно сабирају два низа a и b, а резултат сабирања смеши се у низ c ( $c[0] = a[0] + b[0]$ ,  $c[1] = a[1] + b[1]$ , ...,  $c[n] = a[n] + b[n]$ ):
- a. Елементи сва три низа су у двострукој прецизности
- b. Заглавље функције (потпрограма) дато је у главни.c фајлу у Z3 folderу
- c. Сва три низа прослеђена су потпрограму као параметри пренети по адреси
- d. Потпрограм као повратну вредност враћа код грешке; грешка наступа уколико дође до прекорачења при сабирању; потпрограм враћа 0 уколико није било грешке, а било шта друго уколико јесте

**Напомена:** скрипу за тестирање можете покренути тако што јој само проследите свој потпрограм, односно .S фајл (**не треба прослеђивати .c фајл**) – ./testiraj.sh моје\_решење.S. Уколико добијете грешку која каže да немате privilegije за извршење скриpte, отворите terminal у folderu где је скрипта за тестирање и унесите команду:

chmod +x testiraj.sh