

ZADATAK 2

C++

Napisati program koji modeluje proces naručivanja hrane pomoću klasa:

Obrok, Meso, Prilog.

Klasa **Meso** ima sledeća polja:

- *vrstamesa* - tipa nabrojivi tip **VrstaMesa** sa mogućim vrednostima: SVINJETINA, JUNETINA, PILETINA, BEZ_MESA
- *gramaža* - tipa int

Za klasu **Meso** treba implementirati sledeće:

- konstruktor bez parametara - postavlja gramažu na 0, vrstu mesa na BEZ_MESA
- konstruktor kopije
- set metode za sva polja
- get metode za sva polja
- slobodnu funkciju koja ispisuje gramažu i vrstu mesa (nabrojivi tip ispisati u tekstualnom formatu)

Klasa **Prilog** i ima sledeća polja:

- *vrstapriloga* - tipa nabrojivi tip **VrstaPriloga** sa mogućim vrednostima: RIZA, POMFRIT, PASULJ, TESTENINA, BEZ_PRILOGA

Za klasu **Prilog** treba implementirati sledeće :

- konstruktor bez parametara – postavlja odabran prilog na BEZ_PRILOGA
- konstruktor kopije
- set metodu za polje
- get metode za polje
- slobodnu funkciju koja ispisuje *vrstapriloga*

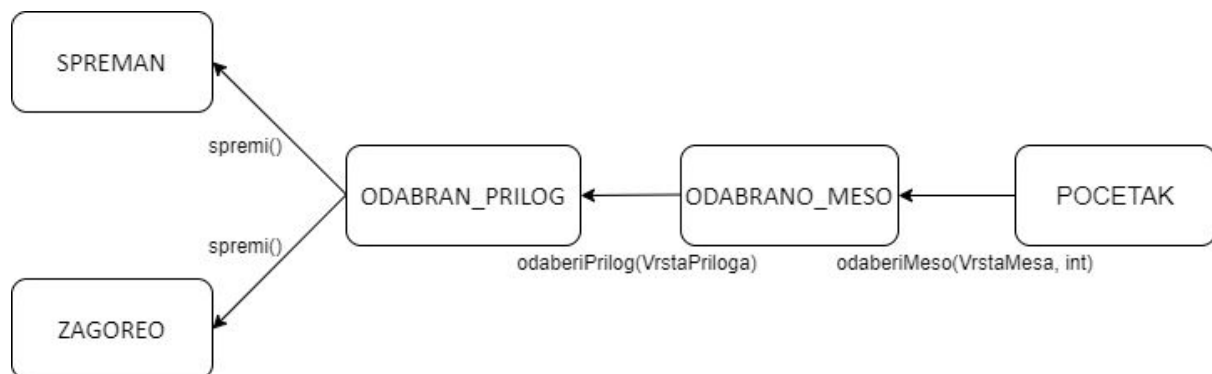
Klasa **Obrok** ima sledeća polja:

- *meso*, tipa **Meso**
- *prilog*, tipa **Prilog**
- *stanje*, tipa nabrojivi tip StanjeObroka sa mogućim vrednostima: POCETAK, ODABRANO_MESO, ODABRAN_PRILOG, SPREMAN, ZAGOREO

Za klasu **Obrok** treba implementirati sledeće:

- konstruktor bez parametara - početno stanje je POCETAK odabira
- konstruktor sa parametrima
- Metoda odaberiMeso(VrstaMesa, int) koja prima vrstu mesa (tipa **VrstaMesa**) i gramažu (tipa **int**) kao parametre. Vrednosti polja *vrstamesa* i *gramaža* unutar polja *meso* treba postaviti na odgovarajuće vrednosti prosleđenih parametara. Gramaža ne sme biti manja od 0g niti veća od 1000g, u suprotnom vraća false i ne prebacuje obrok u stanje ODABRANO_MESO.

- Metoda `odaberiPrilog(VrstaPriloga)` koja prima vrstu priloga (tipa **VrstaPriloga**) kao parametar. Vrednost polja *vrstapriloga* unutar polja *prilog* postaviti na prosleđenu vredost.
- Metoda `spremi()` koja u 90% slučajeva uspešno sprema obrok i prelazi u stanje SPREMAN, dok u 10% slučajeva dolazi do toga da obrok zagori i prelazi u stanje ZAGOREO. Za slučajan izbor procenada koristiti funkciju `rand()`. (`rand()%10` daje broj između 0 i 9, u slučaju da je broj 0, pretpostaviti da je obrok zagoreo, inače da je uspešno spreman)
- slobodnu funkciju za ispis svih polja unutar obroka (kao i njegovih komponenti - *meso, prilog i stanje*)



Napomene:

- U funkciji **main()** testirati svaki implementiran metod. Sve sto nije testirano **ne donosi** bodove.
- Program se mora uspešno kompajlirati, u suprotnom možete imati najviše jedan bod.
- Funkcija `rand` se nalazi u sistemskoj biblioteci `<cstdlib>`
- Postavljanje seed-a funkciji `rand` moguće je obaviti pozivom `srand(time(NULL))`
- Funkcija `time` se nalazi u sistemskoj biblioteci `<time.h>`