

## Vježba 4. – literali, closure, currying

### Upute za rješavanje lab. vježbi

Današnju vježbu pod nazivom *closure.js* dodajte u projekt kako bi sve imali na jednom mjestu. Kod ispisa vrijednosti, poruka ili povratne vrijednosti funkcije potrebno je koristiti literale. Za polaganje vježbi potrebno je riješiti **barem tri zadatka** po želji, a preostale zadatke do sljedećih vježbi.

### Zadaci

1. Kreirati niz koji će sadržavati deset [slučajno](#) generiranih brojeva u intervalu od 100 do 200, [zaokruženi](#) na najbliži cijeli broj. Sortirati elemente niza od manjeg prema većem funkcijom `sort()`. Obratiti pozornost kod [sortiranja](#) numeričkih i tekstualnih elemenata.
2. Omogućiti unos brojeva dok se ne unese 0. Zbrojiti parne i neparne brojeve koristeći se [closure](#) funkcijom [arrow](#) zapisa. Zabranjeno je koristiti nizove.
3. Kreirati closure funkciju koja vraća poruku „*Programirati nije teško*“.
4. Prepiši programski kôd i objasni što se događa.

```
function sum(a){
  return (b, c) => {
    return a * b * c
  }
}
let x = sum(10);
console.log(x(3,12));
// ili
console.log(sum(10)(3,12));
```

5. Korištenjem currying i arrow zapisa, omogućiti pozivanje funkcije s promjenjivim brojem argumenata u formatu (a)(b)(c)... koja bi trebala vratiti umnožak pozitivnih brojeva manjih od 20.