

- 1) Proszę napisać funkcję
`double *vec(double xstart, double xend, int n)`, która zwróci wektor, zawierający `n` wartości `double`, rozłożonych równomiernie pomiędzy `xstart` a `xend` (włącznie).
- 2) Proszę napisać funkcję, która posortuje niemalejąco znaki w napisie podanym jako argument: `void ssort(char *string)`.
- 3) Proszę napisać program, która policzy małe litery alfabetu angielskiego w pliku. Nazwę pliku podajemy jako argument wywołania. Jeśli program zostanie wywołany bez argumentu, powinien liczyć litery w `stdin`.
- 4) Proszę zdefiniować strukturę danych opisującą studenta: imię, nazwisko, nr albumu. Proszę napisać funkcję wyszukującą studenta wg numeru albumu w tablicy takich struktur. Funkcja powinna zwracać wskaźnik do znalezionej struktury lub `NULL`, jeśli nie znalazła pasującego studenta.
- 5) Proszę wyjaśnić znaczenie reguły z pliku `makefile`:

```
runner: main.o sorter.o tester.o util.o
$(CC) -o runner $^ -lm
./runner test_data -o result
diff result good_sample
```
- 6) Proszę napisać program obliczający sumę liczb podanych jako argumenty wywołania. Jeżeli nie podano argumentów, program powinien zwracać kod błędu (równy 1).
- 7) Proszę napisać funkcję obliczającą średnią z elementów przechowywanych w liście liniowej:

```
typedef struct e {
    double value;
    struct e * next;
} * list_t;
```

Jeśli lista jest pusta, funkcja powinna zwracać zero.