

ALPR – Cvičenie 7

Na tomto cvičení si precvičíte prácu s knižnicami. Naučíte sa knižnice **string**, **stdlib**, **stdio** a **ctype**.

Vaše RPG rozšírite o RNG, čo pridá vašej hre replayabilitu.

Screenshotty odovzdávajte do MS Teams do zadania „cvičenie 7“ vo formáte PDF spolu (použite napríklad Word a následný export do .pdf je možné urobiť v záložke **Súbor->Uložiť ako**).

Do Gemu sa prihlasujte **len pomocou STU konta v inkognito okne prehliadača**, inak to nemusí fungovať správne:

<https://gemini.google.com/gem/1GEa8qnZVG862Zd7WA69BiGyAzXMODgvW?usp=sharing>

Príklad 1

Program načíta dve vety, rozdelí ich na jednotlivé slová, a zistí, na ktorých pozíciách sa nachádzajú rovnaké slová v oboch vetách. Tieto slová sa uložia do poľa, ktoré bude narastať s tým ako budú pribúdať nové slová. Vypíšte slová ktoré sú rovnaké a ich pozície. Využite funkcie **strtok_r** a **strcmp**.

Ak zadáme:

„Programovanie v C je ťažké.“

„Programovanie v Pythone je jednoduché.“

Program vypíše:

Programovanie 0

v 1

je 3

Príklad 2

Vytvorte program v jazyku C, ktorý od používateľa načíta údaje: meno, priezvisko, email, heslo a overenie hesla. Program overí:

- skontroluje, či email obsahuje znak '@'
- skontroluje, či heslo má minimálne 8 znakov
- overí, či heslo obsahuje aspoň jedno malé písmeno, jedno veľké písmeno a jedno číslo
- porovná heslo a overenie hesla (musia byť rovnaké)

Program vypíše celé meno, dĺžku celého mena, informáciu, či je email platný, informáciu, či heslo spĺňa požiadavky a informáciu, či sa heslá zhodujú.

Ak všetky údaje splnili požiadavky, program ich zapíše do súboru oddelené tabulátormi.

Program najskôr napíšte bez použitia knižnice **<ctype.h>**, potom kód zakomentujte a vyskúšajte si použitie tejto knižnice, hlavne funkcie **isdigit**, **islower**, **isupper**.

https://www.w3schools.com/c/c_ref_ctype.php

Príklad 3

Program načíta od používateľa počet čísel **n**, následne načíta tieto čísla z klávesnice a vypočíta ich priemer. Po vypísaní priemeru sa program opýta používateľa, či chce pokračovať a zadať ďalšie čísla.

Ak sa používateľ rozhodne pokračovať, zadá nový počet čísel. Program následne realokuje pamäť poľa na väčšiu veľkosť tak, aby sa zachovali už existujúce hodnoty, a nové čísla sa pridajú na koniec pôvodného poľa. Potom sa opäť vypočíta priemer zo všetkých doteraz zadaných čísel.

Príklad 4 – implementácia RNG

Tento príklad si dobre odložte, budete na ňom pracovať aj na nasledujúcich cvičeniach a budete ho rozširovať o rôzne funkcionality.

Na predchádzajúcom cvičení ste upravili funkciu **choseYourAdventure** na to aby ste cestu načítavali zo súboru a začali ste zapisovať históriu vašej cesty do súboru.

Dneska implementujete RNG do vašej hry. Príšery a hrdina dostanú **hitChance**, čo bude reprezentovať percentuálnu šancu na zásah hrdinu a nebudú robiť presný DMG, ale v nejakom rozmedzí. Napríklad, ak goblin do teraz robil 20dmg stále, teraz bude robiť 15-25 s 80% šancou na zásah.

Upravte takýmto spôsobom útoky všetkých príšer a aj hrdinu.

Nepoužívajte štruktúry.