

Python	<p>Tytuł: Spider-snake</p> <p>Treść:</p> <p>Przy użyciu Python'a i biblioteki FastApi (https://fastapi.tiangolo.com) stwórz prostą aplikację sieciową</p> <p>Wymagania:</p> <ul style="list-style-type: none"> • Program używa Pythona i FastApi • Program wyświetla w przeglądarce pod adresem localhost pewien tekst • Program powinien obsługiwać jednego GETa zwracając rzeczony tekst oraz jednego POSTa, który będzie określał treść rzeczonego tekstu przez zawartość body
Pseudo-assembler	<p>Zrealizować program dekodujący wybrany podzbiór kodów kreskowych zgodnych ze specyfikacją Code 128.</p> <p>Opis kodów kreskowych Code 128</p> <p>Struktura kodu kreskowego Code 128 została opisana w rozdziale 3.9.</p> <p>Wejście</p> <p>Dane wejściowe dla programu znajdują się w 24 bitowym pliku BMP[5] bez kompresji. Pliki nie spełniające w/w wymagania nie podlegają przetwarzaniu. Kreski narysowane są równoległe do pionowej krawędzi kolorem czarnym, tło ma kolor biały. W obrazie nie występują żadne zniekształcenia (np. przekos, niedokładności druku itp.).</p> <p>Wyjście</p> <p>Program powinien wypisywać wyniki dekodowania kodu kreskowego na standardowym wyjściu.</p> <p>Program dekoduje znaki z zestawu Code Set AUwagi:</p> <p>1. W programie należy zaimplementować w assemblerze Intel'a następującą funkcję:</p> <pre>int Decode128(unsigned char *image, char *text, int xsize, int ysize, int scanline);</pre> <p>Wartość zwracana: 0 w przypadku powodzenia lub kod błędu w przeciwnym przypadku.</p> <p>Parametry:</p> <ul style="list-style-type: none"> -unsigned char *image – wskazanie na bufor zawierający obraz (bez nagłówek BMP) -char *text – wskazanie na bufor zawierający wyniki dekodowania -int xsize – rozmiar poziomy obrazu w pikselach -int ysize – rozmiar pionowy obrazu w pikselach -int scanline – numer linii w obrazie, w której dokonywane jest dekodowanie.

	<p>2. Odczyt i zapis danych z plików realizowany jest na poziomie języka C.</p> <p>3. Alokacja pamięci na bufory realizowana jest na poziomie języka C.</p> <p>4. Składnia wywołania programu: decode128 plik.bmp nr_linii_w_obrazie</p> <p>5. Rozpoznawanie kodów kreskowych on-line: http://online-barcode-reader.inliteresearch.com/default.aspx</p> <p>6. Generowanie kodów kreskowych on-line: http://www.barcode-generator.org/</p>
Algorytmiczne	<p>Tytuł: My Precious</p> <p>Treść:</p> <p>W dawnych czasach, gdy komputery miały przycisk "Turbo", a do internetu łączyło się wybierając numer 0-202122, pojawiła się gra komputerowa "Rocks'n Diamonds" gdzie gracz wcielał się w rolę eksploratora zbierającego diamenty i uciekającego przed spadającymi mu na głowę kamieniami. Spróbujmy zrobić uproszczoną wersję gry dla megaleniwego eksploratora. Załóżmy, że postać chodzi po prostokątnej planszy $m \times n$ zaczynając od lewego górnego pola, a kończąc w prawym dolnym. Leniwy eksplorator może poruszać się tylko w dół bądź w prawo, czyli przechodząc z kolumny 1 do 2, nie ma już możliwości powrotu do kolumny 1.</p> <p>Na planszy rozmieszczone są diamenty, a celem eksploratora jest zebranie ich największej liczby.</p> <p>Dane wejściowe:</p> <p>W pierwszej linii wejścia znajdują się dwie liczby m i n definiujące rozmiar planszy. W drugiej linii wejścia znajduje się liczba d określająca liczbę diamentów. W kolejnych d liniach znajdują się koordynaty poszczególnych diamentów na planszy</p> <p>3 3 3 0 1 1 2 2 0</p> <p>Dane wyjściowe:</p> <p>W pierwszej linii wyjścia znajduje się liczba zdobytych diamentów W drugiej linii wyjścia znajduje się droga przebyta przez eksploratora w formie ciągu znaków D (dół) i P (prawo) określająca kolejne ruchy</p> <p>2 PPDD</p> <p>Wymagania:</p> <ul style="list-style-type: none"> • Załóż, że plansza nie będzie większa niż 20x20 • Jeśli jest kilka możliwych rozwiązań, wystarczy wypisanie jednego • Możesz założyć numerowanie wierszy/kolumn od 0
Nieszablonowe	<p>Tytuł: mov Hello, World</p> <p>Treść:</p>

	<p>Napisz program typu „Hello world” w języku assemblera NASM64. Program ma wypisywać hasło HACKATHON</p> <p>Dane wejściowe: Brak</p> <p>Dane wyjściowe: Jedna linia z hasłem „HACKATHON”.</p> <p>Wymagania:</p> <ul style="list-style-type: none"> • Konieczność użycia assemblera NASM64
Otwarte	<p>Tytuł: Statystyki tekstu</p> <p>Treść: Stwórz program do zliczania statystyk tekstu. Zaimplementuj między innymi:</p> <ul style="list-style-type: none"> • Zliczanie ilości znaków • Zliczanie ilości słów • Zliczanie ilości wierszy • Inne dodatkowe statystyki (wedle pomysłu)