

Informacja nt. realizacji mini-projektów, przesyłania raportów oraz przeglądu mini-projektów

1. Mini-Projekt

1.1. Algorytm rozwiązania zadania może wykorzystywać wyłącznie:

- podstawowe instrukcje i struktury sterujące: przypisanie wartości, porównanie, pętla, rekurencja,
- podstawowe operatory arytmetyczne i logiczne.

1.2. Możliwe jest definiowanie procedur i funkcji pomocniczych.

1.3. Przykładowa procedura rysowania zależności czasu realizacji algorytmu od rozmiaru danych wejściowych jest dostępna w UBIK - plik " wykres_czas_realizacji_algorytmu.sce".

1.4. Plik "generator tablic" zawiera funkcje pomocnicze generujące wektory i macierze o wartościach całkowitych losowych, uporządkowane i nieporządkowane; funkcje te można używać w projektach.

2. Raport

2.1. Wzór raportu jest udostępniony w UBIK.

2.2. Raport z projektu wraz kodem algorytmu należy przesłać z konta uczelnianego nie później niż 4 doby (96 godzin) przed planowanym terminem prezentacji projektu (nie dotyczy osób prezentujących projekt w postaci filmiku wideo).

2.4. Wiadomość powinna być zatytułowana zgodnie z przykładem:

IZ_BAL_Iksinski.Jan.IZ01P11.Projekt30

2.5. Pliki z raportem i kodem powinny być nazwane zgodnie z przykładem:

IZ_BAL_Iksinski.Jan.IZ01P11.Projekt30.doc

IZ_BAL_Iksinski.Jan.IZ01P11.Projekt30.pdf

IZ_BAL_Iksinski.Jan.IZ01P11.Projekt30.sce

3. Przegląd mini-projektów ("obrony")

3.1. Terminy obron będą ustalane na zajęciach.

3.2. Obrona projektu trwa maksymalnie 5 minut.

3.3. Podczas obrony można posiłkować się przygotowaną prezentacją (wskazane jest przygotowanie kopii prezentacji w formacie pdf na wypadek nie działania zasadniczej prezentacji).

4. Struktura prezentacji pracy

Krótkie (ok. 5 minut) publiczne wystąpienie, służące przedstawieniu projektu w sposób możliwie atrakcyjny, oraz udzieleniu odpowiedzi na pytania dotyczące projektu.

4.1. Przedstawienie się prelegenta.

4.2. Przedstawienie treści zadania projektowego.

4.3. Omówienie algorytmu rozwiązującego zadany problem, w szczególności:

- wymienienie stosowanych metod algorytmicznych (jeśli były świadomie wykorzystywane);

- wymienienie używanych struktur danych (zmienne, wektory, macierze, drzewa, etc.);
 - wymienienie wykorzystywanych w algorytmie struktur sterujących.
- 4.4. Omówienie schematu blokowego (jeśli jest bardzo skomplikowany, to warto rozbić go na części).
- 4.5. Przedstawienie kodu SCILAB (w przypadku skomplikowanego kodu warto go rozbić na funkcje lub procedury).
- 4.6. Omówienie złożoności algorytmu w najgorszym przypadku (wskazanie operacji dominującej, obliczenie funkcji złożoności oraz rzędu złożoności).
- 4.7. Zademonstrowanie działania kodu w "realu", dla przykładowych danych (mogą być losowe).
- 4.8. Przedstawienie wykresu zależności czasu realizacji algorytmu od rozmiaru danych wejściowych oraz odniesienie go do oszacowanego wcześniej rzędu złożoności.
- 4.9. Wnioski ("udało się...", "mój algorytm...", "można by..."), odpowiedź na ewentualne pytania słuchaczy.

5. Obrona projektu w formie filmiku wideo

Uwagi:

- nagranie nie musi być przeprowadzone w jednym ujęciu, może być montowane z wielu części (to daje dużą przewagę względem prezentowania w "realu");
- czas trwania filmiku powinien się mieścić w przedziale 4,5-5,5 minuty;
- sugerowana rozdzielczość to 460p, ewentualnie 720p (HD);
- maksymalna wielkość pliku wideo: ok. 100MB;
- nie-niszowy format (np. mp4).

Ostateczny termin przesyłania prezentacji projektu w formie filmiku wideo upływa **02/02/2020**
(wraz z prezentacją należy przesać raport i pliki projektu!).

Zachęcam do konsultowania projektów.

Jestem dostępny w poniedziałki 16:00-18:00 oraz czwartki 16:00-18:00, pok. N125 na ul. Newelskiej.

W innych dniach i porach spotkanie możliwe po wcześniejszym umówieniu się.