1. **Tytuł projektu:** bigOS → system operacyjny grupy L5b
2. **Język programowania:** najbardziej zjebany → dżawa
3. **Wykonawcy** 
   1. Wojciech Kulczak → zabezpieczenia →
   2. Anna Lenhardt → Interpreter →
   3. Konrad Kęciński → Interfejs →
   4. Jonasz Łapiński → Zarządzanie procesorem metoda piorytetową →
   5. Maciej Sobkowiak → Zarządzanie pamiecią operacyjną metodą obszarów ciąglych przesównych →
   6. Piotr Kwiatkowski → Zarządzanie procesami →
   7. Rafał Ewiak → Mechanizmy synchronizacyjne: semafory całkowitoliczbowe →
   8. Weronika Kowalska → Zarządzanie plikami za pomocą tablic FAT →
   9. Wojciech Lulek → Komunikacja międzyprocesowa za pomocą komunikatów tekstowych →
4. **Nazwy i przeznaczenie podstawowych struktur danych w danym module** (min. 1, maks. 3)
   1. …
   2. …
   3. ...
   4. …
   5. …
   6. …
   7. …
   8. Integer[] FAT – tablica FAT,
   9. char[] data – dysk posiadający 32 bloki po 32 bajty,
   10. List<File> root – lista zawierająca pliki, katalog główny, **//to można wywalić jak komuś zabraknie numerków**
   11. …
   12. Queue<Message> messages\_queue - kolejka wiadomości otrzymanych przez dany proces
   13. …
   14. …
   15. …
   16. Map<Integer, Process> processes → przechowywanie orginałów PCB,
   17. Vector<String> taken\_names → przechowywanie zajętych nazw w celu zabezpieczenia przed powstaniem kilku procesów o tej samej nazwie,
   18. Map<String, Integer> names → mapa, w której kluczem jest nazwa procesu, a wartością jego PID, ułatwia odwoływanie się do konkretnych procesów,
5. **Nagłówki (w języku C++ lub Java) podstawowych procedur/funkcji realizujących dany moduł** (min. 1, maks. 3)
   1. …
   2. …
   3. ...
   4. …
   5. …
   6. …
   7. …
   8. public static boolean create(String name, String user);
   9. public static boolean write(String name, String data);
   10. public static String read(String fileName, int from, int howMany);
   11. …
   12. public boolean send\_message(int receiverPID, String text)
   13. public boolean send\_message(int receiverPID, int size, int addres)
   14. public boolean read\_message(int addres)
   15. …
   16. …
   17. public static void create\_process(String name, String file\_name, int priority),
   18. public static void delete(int PID) oraz public static void delete(String name),
   19. public static void add\_to\_CPU(int PID),
6. **Jednozdaniowe uwagi dotyczące interfejsów do pracy krokowej i współdziałania z innymi modułami** (maks. 3 w danym module)
   1. …
   2. …
   3. …
   4. …
   5. …
   6. Umożliwiam wyświetlenie PCB wszystkich oraz konkretnych procesów, dodatkowo metody działające w ramach funkcjonalności modułu opisują swoje działania w konsoli w trakcie wykonywania.
   7. …
   8. Możliwość wyświetlenia bieżącej zawartości dysku, wektora bitowego, tablicy FAT oraz FCB pliku wraz ze stanem semafora dla danego pliku.
   9. Możliwość wyświetlenia zawartości kolejki wiadomości otrzymanych od innych procesów oraz stanu semafora regulującego odczytywanie wiadomości.
   10. …
   11. …
   12. …