

Pytanie	Odpowiedź	
(pt. 12.30) Co robi flaga O_TRUNC? (pt. 12.30) Jakie są prawidłowe wartości zwracane przez funkcję w przypadku wystąpienia błędu? (śr 17:00) co to jest errno (śr 17:00) wymienić standardowe deskryptory plików (śr 18:30) Czy program jednomodułowy może składać się z wielu plików *.c? (czw 17:45) Ile jest instancji zmiennej errno (coś tam coś tam, chyba jak jest kilka wątków i procesów) (czw 17:45) Do czego służy flaga O_CREAT w open(2) jak funkcja sygnalizuje błąd (pt 15:45) Kiedy wywoływane są cele PHONNY w make?	Czyści plik (zastępuje jego zawartość bitowymi zerami) makro przechowująca kod ostatniego błędu 0 stdin, 1 stdout, 2 stderr może, ale jeden z plików musi inkludować inne (coś takiego było na wiki) Tyle ile jest wątków w procesie tworzy plik, jeżeli plik o danej ścieżce nie istnieje zwraca -1, NULL lub SIG_ERR na końcu, zawsze	
WEJŚCIÓWKA 2 - 12-16.03		
Pytanie	Odpowiedź	
(pt. 12.30) W jakiej strukturze przechowywane są informacje o wykonywanych operacjach asynchronicznych? (śr 17:00) jak za pomocą funkcji lseek sprawdzić rozmiar pliku (śr 18:30) Co to znaczy, że operacja nie jest zsynchronizowana? (śr 17:00) co robi funkcja stat (śr 18:30) Wymień co najmniej jeden typ pliku, w którym nie można ustawić wskaźnika. (czw 17:45) Jak lseekiem sprawdzić aktualny wskaźnik pliku (czw 17:45) Co to znaczy, że operacja(e) I/O są synchroniczne (pt 14:00) jak za pomocą funkcji lseek sprawdzić rozmiar pliku (pt 14:00)-czym jest funkcja stat (pt 15:45)Jak za pomocą lseek zobaczyć gdzie jest offset? jak nazywa się struktura z informacją o operacji asynchronicznej (cos w tym stylu) (pt 15:45)Czy można zrobić hard link pliku, który znajduje się na innym urządzeniu? Uzasadnij	Identyfikator pliku wyrażony nieujemną liczbą całkowitą. Używają go funkcje, które operują na plikach (np. read, write) ??? aiocb lseek(f, 0, SEEK_END) fd - deskryptor pliku to znaczy że kończy się w programie, ale niekoniecznie znaczy to fizyczne zakończenie operacji zwraca informacje (metadane) o pliku gniaźdo, kolejka lseek(f, 0, SEEK_CUR) fd - deskryptor pliku Wyjście z funkcji systemowej nastąpi po zakończeniu fizycznej operacji I/O lseek(f, 0, SEEK_END) fd - deskryptor pliku zwraca informacje (metadane) o pliku lseek(f, 0, SEEK_CUR) aiocb nie, bo prowadziłoby to do niejednoznacznych odwołań + po odmontowaniu urządzenia mielibyśmy link donikąd	
WEJŚCIÓWKA 3: 19 - 23.03		
Pytanie	Odpowiedź	
(pon. 8:30) Różnica między stat i lstat (pon. 8:30) Jakie wartości zwraca funkcja fork? (wt. 17:15) co wykona się(jak zadziałają) dla A) polecenie1 & polecenie2, a co B) dla polecenie1 polecenie2? (śr. 17:00) co zwraca funkcja exec (śr 17:00) jakie warunki musi spełniać plik żeby go można było usunąć (śr 18:30) Zakres priorytetów (śr 18:30) Funkcje, które wyświetlają metadane o dowiązaniu symbolicznym i o pliku na podstawie dowiązania symbolicznego (czw. 17:45) Jaka funkcja można sterować deskryptorem pliku? (czw. 17:45) Czy execve() dziedziczy PID i deskryptory po procesie go wywołującym (pt. 12:30) Co zrobić żeby deskryptor pliku został zamknięty po wywołaniu exec()	stat() - metadane o pliku; lstat() - meta dane o dowiązaniu symbolicznym PID dziecka dla rodzica; 0 dla dziecka A) wykona się polecenie1, jeżeli: - zwróci błąd to nie przejdzie do polecenie2 - wykona się bez błędu to przejdzie do polecenie2 B) wykona się polecenie1, jeżeli: - zwróci błąd to przejdzie do polecenie2 - wykona się bez błędu to nie przejdzie do polecenie2 jak się wykona poprawnie to wgl nie wraza, bo zasadniczo nie ma jak; jak źle to -1 1. Nie ma w systemie plików dowiązań do danych i-node'a; 2. Zaden proces nie ma otwartego tego pliku (tj. nie ma deskryptora odpowiadającego temu plikowi) -20 do 19 stat() - metadane o pliku; lstat() - meta dane o dowiązaniu symbolicznym fcntl -> (man ref: fcntl dokonuje jednej z wielu różnych operacji na fd. Wykonywana operacja zdeterminowana jest przez cmd; gdzie cmd jest argumentem fcntl) na pewno nowy proces dziedziczy pid w sensie no, ma ten sam pid co proces, który go wywołał; a jeżeli chodzi o deskryptory, to nie są one zamykane, man ref: By default, file descriptors remain open across an execve(). File descriptors that are marked close-on-exec are closed; see the description of FD_CLOEXEC in fcntl(2). chodzi o close-on-exec ustawione przez fcntl SKOPIOWANE Z WIKI: Po wywołaniu fork oba procesy (przodek i potomek) mają wspólny segment danych składający się ze stron. Dopóki strona nie ulega zmianie to jest wszystko ok. Gdy któryś z procesów będzie próbował coś zapisać na stronie, to będzie ona skopiowana i każdy proces będzie miał własną wersję. Identyfikacja, za wyjątkiem dowiązania symbolicznego: stat() - metadane o pliku; lstat() - meta dane o dowiązaniu symbolicznym PID dziecka dla rodzica; 0 dla dziecka fcntl -20 do 19	
WEJŚCIÓWKA 4: 26.03 - 06.04		
Pytanie	Odpowiedź	
(pon. 8:30) Ile procesów będzie istniało po wywołaniu fork(); fork(); fork(); (pon. 8:30) Różnica między procesem i wątkiem (wt. 17:15) Czego używamy do synchronizacji wątków (wt 18:45) Argumenty funkcji pthread_create() (śr 17:00) różnica między mutexem a ryglem odczytu-zapisu (read-write lock) [1] (śr 17:00) Ile procesów utworzy wywołanie 3 razy fork() (pt. 12:30) fork(); fork(); fork(); pytanie - ile razy program wywoła fork 1 raz (pt. 12:30) funkcja tworząca wątki - jakie przyjmują wartości (pt. 14:00) 1 wątek master, 3 wątki poboczne, jeden z pobocznych wywołuje fork()- ile będzie wątków w takim procesie? (pt. 14:00) Co to są bariery? XD (pt 15:45) 1 wątek master, 3 wątki poboczne, w procesie są zmienne globalne, jeden z pobocznych wywołuje fork()- ile będzie wątków w takim procesie? (czw. 17:45) 1 wątek master, 3 wątki poboczne, jeden z pobocznych wywołuje fork()- ile będzie wątków w takim procesie + nazwy tych wątków (czw. 17:45) warunki na mutex aby był jako argument pthread_cond_wait()	8 wątki współdzielą pamięć, a procesy nie mutexów pthread_1,"thread, const pthread_attr_t *attr, void (*start_routine)(void)", void *arg RWLOCK wyznacza blokadę zapisu i odczytu (jeśli ktoś blokuje zapis, to nie można ani czytać ani pisać, a jeśli odczyt to można czytać lecz nie można pisać) https://en.wikipedia.org/wiki/Readers%E2%80%93writer_lock 7 2 pthread_1,"thread, const pthread_attr_t *attr, void (*start_routine) (void)", void *arg https://it-ebooks.com/wiki/POSIX_Threads/Synchronizacja_m%C4%99dzy_w%C4%85kamiBariery 2 (chyba główny i poboczny??) Mutex musi być zaryglowany, gdy funkcja jest wywoływana	mutex, cond var (ewentualnie cancelacji) : pthread_mutex_lock, pthread_mutex_unlock, pthread_cond_wait, pthread_cond_signal (pthread_cancel)
KOŁOKWIUM 1		
Pytanie	Odpowiedź	
Pytania powtórzone z wiki		nk wrzuci pytania z kolosa
WEJŚCIÓWKA 5		
Pytanie	Odpowiedź	
(pon. 8:00) Różnica między lseek dla zwykłych plików i łącz komunikacyjnych (pon. 8:00) Kiedy należy stosować dodatkowe mechanizmy synchronizacji podczas komunikacji międzyprocesowej wt(17:15) Jakże wartości liczbowe przyjmują deskryptory standardowego wejścia? wt(17:15) Jak zastąpić funkcję dup przy pomocy funkcji fcntl? (wt 18:45) Opis działania dup (wt 18:45) Różnica lseek dla pliku, a łącz (czw. 17:45) dup2 <=> fcntl (czw. 17:45) Jak można zasygnalizować znak końca pliku w łączu nienazwanym? (pt. 12:30) dup2 za pomocą fcntl (pt. 12:30) czym się różni dup2 od dup2 za pomocą fcntl	w łączach komunikacyjnych lseek się nie używa gdy komunikacja odbywa się w dwie strony za pomocą funkcji pipe 0-STDIN, 1-STDOUT, 2-STDERR fcntl(f, F_DUPFD, 0); Pobiera deskryptor o najmniejszym dostępnym numerze - jeśli wiemy który fd jest otwarty, możemy sterować tym co dup przekazuje dla łącz komunikacyjnych nie używa się lseek. close(f2); fcntl(f, F_DUPFD, f2); zamknąć wszystkie deskryptory piszące, także te w procesie czytającym oraz pipe musi być pusty	
WEJŚCIÓWKA 6		
Pytanie	Odpowiedź	
wt(17:15) Ile SUMARYCZNIE deskryptorów do stworzonego łącz komunikacyjnego będzie istnieć po wywołaniu funkcji fork zakładając, że łącz zostało stworzone w procesie w którym wywołano fork przed tym wywołaniem oraz było to pierwsze wywołanie tej funkcji? wt(17:15) Jak zagwarantować aby funkcja semget utworzyła unikatowy obiekt IPC? śr(17:00) Ile SUMARYCZNIE deskryptorów do stworzonego łącz komunikacyjnego będzie istnieć po wywołaniu funkcji fork zakładając, że łącz zostało stworzone w procesie w którym wywołano fork przed tym wywołaniem oraz było to pierwsze wywołanie tej funkcji? śr(17:00) Jak zagwarantować aby funkcja semget utworzyła unikatowy obiekt IPC?	4 ustawić flagę IPC_PRIVATE 4 ustawić flagę IPC_PRIVATE	Powtórzyło się :D
WEJŚCIÓWKA 7		
Pytanie	Odpowiedź	
pon(8:00) jakimi 2 funkcjami zaryglować plik pon(8:00) jak stworzyć kolejkę FIFO poleceniem w terminalu (wt. 17:15) Jak program uruchamiany przy pomocy konsoli pozwala usunąć stworzone łącze komunikacyjne w systemie V7? (wt. 17:15) Na czym polega zakleśczenie procesów? (pt. 12:30) Ryglowanie obowiązkowe i zalecane różnica (pt. 12:30) Jak stworzyć kolejkę FIFO poleceniem w terminalu	fcntl, lockf mkfifo ipcrm sytuacja, w której co najmniej dwie różne akcje czekają na siebie nawzajem, więc żadna nie może się zakończyć. "Advisory locking requires cooperation from the participating processes." "Mandatory locking doesn't require cooperation from the participating processes." mkfifo	
WEJŚCIÓWKA 8		
Pytanie	Odpowiedź	
(pon. 8:00) Funkcja inicjująca semafor nienazwany (pon. 8:00) Co to jest zadanie? (śr. 18:30) Co robi funkcja ipc (śr. 18:30) Terminal kanoniczny i niekanoniczny (czw. 17:45) Podaj nazwę oraz parametry funkcji zwracającej atrybuty terminalu (pt. 12:30) Co się dzieje przy nawiązaniu połączenia przez serwer	sem_init grupa procesów Wyświetla informacje o obiektach IPC kanonicznym – tryb standardowy; linie są buforowane, terminal obsługuje klawisze specjalne (kil, erase itp.); niekanonicznym – linie nie są buforowane, dane wejściowe są dostępne na poziomie bajtów, czyli tak samo jak w DOS. tcgetattr ? 1. socket, 2. bind, 3. listen, 4. accept	
WEJŚCIÓWKA 9		
Pytanie	Odpowiedź	
(pon. 8:00) nawiązywanie połączenia przez serwer	1 - socket, 2 - bind, 3 - listen, 4 - accept	

(pon. 8:00) Jaką rolę pełni operacja bind?	Powiązanie numeru portu z dyskryptorem gniazda.	
Typy komunikacji przez sockety	TCP, UDP	
3 typy adresow (po dopytaniu Gierdziewicz powiedział że chodzi o protokoły) xdd	ipv4, ipv6, mac	
klient czy serwer korzysta z gniazdowego deskryptora pliku	oba xd	
wejściówka 10 /c to chyba dalej wejściówka nr 9, czyż nie? /eoc		
Co robi listen?	nasłuchuje na socketcie	
Jak sprawdzić adres źródłowy komunikatu w komunikacji bezpołączeniowej?	adres źródłowy jest argumentem funkcji recvfrom()	

[illegible]

[illegible]

[illegible]

[1] Jak się odmienia "rygłem"?
Ale to jest rygiel do drzwi, czy jak?
-to read-write lock