**Crontab - opis**

**crontab** - tabela programu [**cron**](https://pl.wikipedia.org/wiki/Cron_(Unix)) posiadająca specjalny format oraz nazwa programu służącego do jej edycji. Etymologia nazwy odnosi się do języka greckiego (χρόνος [*czyt. chronos*] - znaczy czas).

Program *crontab* wywoływany z linii poleceń z opcją -l wyświetla aktualną tabelę użytkownika. Opcja -r usuwa ją, a -e pozwala na jej edycję.

Tabele cronu właściwe dla danych użytkowników znajdują się zwykle w katalogu /var/spool/cron/crontabs (np. jest tak w systemach [HP-UX](https://pl.wikipedia.org/wiki/HP-UX) i [Solaris](https://pl.wikipedia.org/wiki/Solaris_(system_operacyjny)), z kolei dystrybucja [Linuksa](https://pl.wikipedia.org/wiki/Linux) [RHEL](https://pl.wikipedia.org/wiki/Red_Hat_Enterprise_Linux) i jej klony używają w tym celu /var/spool/cron).

Niektóre dystrybucje Linuksa mają dodatkową tabelę systemową umieszczoną w pliku /etc/crontab. Od zapisów tam umieszczonych zależy sposób obsługi skryptów z katalogów /etc/cron.\*:

/etc/cron.hourly - skrypty wykonywane co godzinę

/etc/cron.daily - skrypty wykonywane codziennie

/etc/cron.weekly - skrypty wykonywane raz w tygodniu

/etc/cron.monthly - skrypty wykonywane raz w miesiącu

W powyższych katalogach należy umieścić plik ze skryptem do wykonania.

**Przykład:**

Przykładowa tabela crontab z systemu [GNU/Linux](https://pl.wikipedia.org/wiki/GNU/Linux) właściwa dla danego użytkownika:

# Używaj /bin/sh do wywoływania komend. Nieważne co jest w /etc/passwd.

SHELL=/bin/sh

#

# Przesyłaj wyjście do użytkownika paul

MAILTO=paul

#

# Użyj następującej wartości zmiennej PATH

PATH=/sbin:/bin:/usr/sbin:/usr/bin:/home/paul/bin

#

# Uruchamiaj się 5 minut po północy, codziennie

5 0 \* \* \* $HOME/bin/daily.job >> $HOME/tmp/out 2>&1

#

# Uruchamiaj się o 14:15 pierwszego dnia każdego miesiąca -- wyjście jest

# przesyłane do paula

15 14 1 \* \* $HOME/bin/monthly

#

# Denerwuj Stefana od poniedziałku do piątku o godzinie 22:00

0 22 \* \* 1-5 echo "Stefanie, jest już 22:00. Gdzie są Twoje dzieci?!" | mail -s "Wybiła 22:00" stefan@example.com

#

# Inne przykłady

23 0-23/2 \* \* \* echo "uruchamiaj 23 min po polnocy, 2, 4 ..., codziennie"

5 4 \* \* sun echo "Uruchamiaj się 5 po 4 w niedziele"

0 10 \* \* 1,3 echo "Uruchamiaj się w każdy poniedzialek i srode o 10.00"

\*/2 \* \* \* \* echo "Uruchamiaj się co 2 minuty"

lub z wykorzystaniem zapisu słownego z wykorzystaniem znaku "@" tzn.:

@reboot echo "System uruchomił się!" -polecenie uruchamiane każdorazowo po uruchomieniu systemu

@yearly echo "Minął kolejny rok!" -odpowiednik zapisu "0 0 1 1 \*"

@annually echo "Minął kolejny rok!" - j.w.

@monthly echo "Minął kolejny miesiąc!" -odpowiednik zapisu "0 0 1 \* \*"

@weekly echo "Minął kolejny tydzień!" -odpowiednik zapisu "0 0 \* \* 1"

@daily echo "Minął kolejny dzień!" -odpowiednik zapisu "0 0 \* \* \*"

@midnight echo "Minął kolejny dzień!"- j.w.

@hourly echo "Minęła kolejna godzina!" -odpowiednik zapisu "0 \* \* \* \*"

Pierwsza część pliku to definicje trzech zmiennych środowiskowych *SHELL*, *MAILTO* i *PATH*. Wartość zmiennej *SHELL* ustala powłokę w której cron będzie uruchamiał polecenia. Wartość zmiennej *MAILTO* określa użytkownika, któremu [pocztą elektroniczną](https://pl.wikipedia.org/wiki/Poczta_elektroniczna) wysłany zostanie raport zawierający [standardowy strumień wyjścia](https://pl.wikipedia.org/wiki/Standardowe_strumienie#Standardowy_strumień_wyjścia_(stdout)) oraz [standardowy strumień błędów](https://pl.wikipedia.org/wiki/Standardowe_strumienie#Standardowy_strumień_błędów_(stderr)), jeśli wykonywane polecenie umieści coś na jednym z nich. Zmienna *PATH* zawiera oddzielone dwukropkami ścieżki, których powłoka używa kolejno do odnalezienia plików wykonywalnych.

Druga część to już właściwa tabela zawierająca informacje o czasie uruchomienia i poszczególnych komendach dla każdego z zadań. W tej części linie nie będące komentarzem zawierają sześć kolumn. Pierwsze pięć określa czas uruchomienia zadania, szósta definiuje komendę, która ma zostać wykonana:

\* \* \* \* \* komenda do wykonania

- - - - -

| | | | |

| | | | +----- dzień tygodnia (0 - 7) (niedziela=0, poniedziałek=1, wtorek=2, ..., niedziela=7) (niedziela może być przedstawiona jako 0 lub 7)

| | | |

| | | +------- miesiąc (1 - 12)

| | |

| | +--------- dzień miesiąca (1 - 31)

| |

| +----------- godzina (0 - 23)

|

+------------- minuta (0 - 59)

**Uwagi:**

* W większości implementacji niedziela może być oznaczona jako 0 lub 7
* Aby uniknąć problemów z uruchomieniem poleceń systemowych, należy w crontabie podawać pełne ścieżki do nich lub ustawić odpowiednią wartość zmiennej *PATH*. Tak samo należy postąpić w przypadku skryptów, które są wywoływane przez [cron](https://pl.wikipedia.org/wiki/Cron).
* Dzień wykonania komendy można wyspecyfikować na dwa sposoby: podając dzień miesiąca lub dzień tygodnia. Jeśli oba pola są ustawione, to komenda wykona się zarówno w ustawiony dzień miesiąca, jak i w ustawiony dzień tygodnia!
* Wartości liczbowe możemy zapisywać w różnych formatach:
  + 1-3 - czyli wartości 1,2,3
  + 0-10/2 - czyli wartość 0,2,4,6,8 i 10 (co druga wartość ze zbioru od 0 do 10)
  + 1,2,5 - czyli wartości kolejno 1,2,5
  + \*/2 - co 2 dozwolona wartość (np. w pierwszej kolumnie będzie to 0,2,4,6...56,58)
  + 1-3,5,6 - czyli 1,2,3 oraz 5 i 6
* Tabela systemowa zawarta w pliku */etc/crontab* posiada nieznacznie inną składnię: pierwsze pięć pól określa czas uruchomienia zadania, szóste pole określa nazwę użytkownika, z którego uprawnieniami zadanie zostanie uruchomione, siódme pole definiuje komendę, która zostanie wykonana.
* Jako specjalny znak traktowany jest "%" (znak procent), który oznacza nową linię. Aby wyłączyć tą funkcjonalność należy wstawić przed niego znak "\"
* Z powodu [zmiany czasu](https://pl.wikipedia.org/wiki/Czas_letni) należy unikać ustawiania zadań które miałyby się wykonać między 2:00 a 3:00 w niedzielę. Takie ustawienie powoduje, że przy automatycznej zmianie czasu na letni zadanie się nie wykona, natomiast przy zmianie czasu z letniego na zimowy wykona się dwa razy (między 2:00 i 3:00 oraz 2a:00 i 3a:00).

**Zadanie**

1. Utwórz skrypt, który utworzy pusty plik tekstowy o nazwie xxx.txt w katalogu domowym zsl. Stosując polecenie **crontab – e** dopisz do tabeli crona uruchomienie tego skryptu po upływie 5-ciu minut od czasu bieżącego.
2. W katalogu domowym /home/zsl utwórz katalog o nazwie Archiwum.
3. Napisz skrypt który wykona archiwizację katalogu Dokumenty z katalogu domowego /home/zsl . Nazwa tego archiwum powinna zawierać bieżącą datę, np. 190312.tar Plik ten powinien zostać umieszczony w katalogu Archiwum.
4. Wpisz do tabeli crona wykonanie tego skryptu raz w tygodniu w czasie trwania zajęć z PrSO
5. Sprawdź działanie harmonogramu zawierającego powyższą archiwizację.
6. Wpisz kilka zdarzeń do tabeli crona, polegających na utworzeniu plików tekstowych zawierających opis zdarzenia. Nazwy tych plików powinny zawierać informację o czasie powstania:

np. plik tworzony w każdą sobotę o godz. 8:00 powinien zawierać tekst ” plik tworzony w każdą sobotę o godz. 8:00". Nazwa tego pliku powinna mieć postać np.: sobota\_23 :

np*. touch /home/bolek/sobota\_$(date +%F)*

Uruchom zdarzenia w następujących warunkach:

* w każdą sobotę o godz. 10:00
* co dwie minuty w ciągu najbliższych 10 minut
* w każdy pierwszy dzień miesiąca o godzinie 12:00
* jednorazowo 10 minut przed zakończeniem dzisiejszych zajęć