Nikhil Patten
11 November 2022
Dr. Moe
ASTR5420

99%

Please show all work. If you collaborate with other students, write their names at the top of your homework. Please also write the total number of hours you spent on this. If you use any code or plotting routines, print those out and attach them to your hand-written solutions.
This took me approximately 10 hours to complete.

1. (15%): Winds of Wolf-Rayet Stars:

x 15/15

(a) **Which stars lose $>$ 50% of their mass and become hydrogen-deficient Wolf-Rayet stars. Assume $\tau = 3$ Myr, $L = L_{Edd}(M/100M_{\odot})$, and $R = 10R_{\odot}(M/30M_{\odot})^{0.6}$ for massive stars with solar**

metallicity.

$$\dot{M}\tau > 0.5M_*$$

$$0.5M_* < \left[2 \times 10^{-7}\ M_\odot\ \mathrm{yr}^{-1} \left(\frac{L}{10^5 L_\odot}\right)^{2.2} \left(\frac{M_*}{30M_\odot}\right)^{-1.3} \left(\frac{T_{eff}}{40000\ \mathrm{K}}\right)^{0.9} \left(\frac{Z}{Z_\odot}\right)^{0.6}\right] 3 \times 10^6 \text{ years}$$

$$0.5M_* < 6 \times 10^{-1} \left(\frac{L_{Edd}}{10^5 L_\odot}\left(\frac{M_*}{100M_\odot}\right)\right)^{2.2} \left(\frac{M_*}{30M_\odot}\right)^{-1.3} \left(\frac{5770\ \mathrm{K}}{40000\ \mathrm{K}}\left(\frac{L_*}{L_\odot}\right)^{0.25}\left(\frac{R_*}{R_\odot}\right)^{-0.5}\right)^{0.9} \left(\frac{Z_\odot}{Z_\odot}\right)^{0.6} M_\odot$$

$$0.5M_* < 6 \times 10^{-1} \left(\frac{1}{10^5 L_\odot}\left(3.2 \times 10^4 L_\odot \frac{M_*}{M_\odot}\right)\left(\frac{M_*}{100M_\odot}\right)\right)^{2.2} \left(\frac{M_*}{30M_\odot}\right)^{-1.3}$$

$$\left(\frac{5770}{40000}\left(\frac{L_{Edd}}{L_\odot}\frac{M}{100M_\odot}\right)^{0.25}\left(\frac{10R_\odot}{R_\odot}\left(\frac{M_*}{30M_\odot}\right)^{0.6}\right)^{-0.5}\right)^{0.9} M_\odot$$

$$0.5M_* < 6 \times 10^{-1} \left(3.2 \times 10^{-1} \frac{M_*}{M_\odot}\left(\frac{M_*}{100M_\odot}\right)\right)^{2.2} \left(\frac{M_*}{30M_\odot}\right)^{-1.3}$$

$$\left(\frac{5770}{40000}\left(\frac{3.2 \times 10^4 L_\odot}{L_\odot}\frac{M_*}{M_\odot}\frac{M_*}{100M_\odot}\right)^{0.25}\left(10\left(\frac{M_*}{30M_\odot}\right)^{0.6}\right)^{-0.5}\right)^{0.9} M_\odot$$

$$0.5M_* < 6 \times 10^{-1} \left(3.2 \times 10^{-3} \frac{M_*}{M_\odot}\left(\frac{M_*}{M_\odot}\right)\right)^{2.2} \left(\frac{M_*}{30M_\odot}\right)^{-1.3}$$

$$\left(\frac{5770}{40000}\left(3.2 \times 10^2 \frac{M_*}{M_\odot}\frac{M_*}{M_\odot}\right)^{0.25}(10^{-0.5})\left(\frac{M_*}{30M_\odot}\right)^{-0.3}\right)^{0.9} M_\odot$$

$$0.5M_* < 6 \times 10^{-1} \left(3.2 \times 10^{-3}\right)^{2.2} \left(\frac{M_*}{M_\odot}\right)^{4.4} \left(\frac{M_*}{30M_\odot}\right)^{-1.3}$$

$$\left(\frac{5770}{40000}\left(3.2 \times 10^2\right)^{0.25}\left(\frac{M_*}{M_\odot}\right)^{0.5}(10^{-0.5})\left(\frac{M_*}{30M_\odot}\right)^{-0.3}\right)^{0.9} M_\odot$$

$$0.5M_* < 6 \times 10^{-1} \left(3.2 \times 10^{-3}\right)^{2.2} \left(\frac{M_*}{M_\odot}\right)^{3.1} \left(\frac{1}{30}\right)^{-1.3} \left(\frac{5770}{40000}\right)^{0.9} (10)^{-0.45} \left(3.2 \times 10^2\right)^{0.225} \left(\frac{1}{30}\right)^{-0.27}$$

$$\left(\left(\frac{M_*}{M_\odot}\right)^{0.2}\right)^{0.9} M_\odot$$

$$0.5M_* < 6 \times 10^{-1} \left(3.2 \times 10^{-3}\right)^{2.2} \left(\frac{M_*}{M_\odot}\right)^{3.1} (30)^{1.3} \left(\frac{5770}{40000}\right)^{0.9} (10)^{-0.45} \left(3.2 \times 10^2\right)^{0.225} 30^{0.27} \left(\frac{M_*}{M_\odot}\right)^{0.18} M_\odot$$

$$0.5M_* < 6 \times 10^{-1} \left(3.2 \times 10^{-3}\right)^{2.2} (30)^{1.3} \left(\frac{5770}{40000}\right)^{0.9} (10)^{-0.45} \left(3.2 \times 10^2\right)^{0.225} 30^{0.27} \left(\frac{M_*}{M_\odot}\right)^{3.28} M_\odot$$

$$M_*^{-2.28} < 1.2 \left(3.2 \times 10^{-3}\right)^{2.2} (30)^{1.57} \left(\frac{5770}{40000}\right)^{0.9} (10)^{-0.45} \left(3.2 \times 10^2\right)^{0.225} \left(\frac{1}{M_\odot}\right)^{3.28} M_\odot$$

$$M_*^{2.28} > \left(1.2 \left(3.2 \times 10^{-3}\right)^{2.2} (30)^{1.57} \left(\frac{5770}{40000}\right)^{0.9} (10)^{-0.45} \left(3.2 \times 10^2\right)^{0.225} \left(\frac{1}{M_\odot}\right)^{3.28} M_\odot\right)^{-1}$$

$$M_* > \left(1.2 \left(3.2 \times 10^{-3}\right)^{2.2} (30)^{1.57} \left(\frac{5770}{40000}\right)^{0.9} (10)^{-0.45} \left(3.2 \times 10^2\right)^{0.225} (M_\odot)^{-2.28}\right)^{-1/2.28}$$

$$M_* > \left(1.2 \left(3.2 \times 10^{-3}\right)^{2.2} (30)^{1.57} \left(\frac{5770}{40000}\right)^{0.9} (10)^{-0.45} \left(3.2 \times 10^2\right)^{0.225}\right)^{-1/2.28} M_\odot$$

$$\boxed{M_* > 43.4M_\odot}$$

Code: try doing it numerically next time

```
In [5]: M_wr = (1.2 *(3.2e-3) **2.2 *30 **1.57 *(5770 /40000) **0.9 *10 **-0.45\
           *(3.2e2) **0.225) **(-1 /2.28)
        # NP Calculating lower limit when WR stars loses half of its mass during MS
        print('M > ' +format(M_wr, '.2E') +' M_o')
        # NP Printing result

        M > 4.34E+01 M_o
```

(b) **Wolf-Rayet stars explode as Type Ib/c supernovae without hydrogen in their spectra. What fraction of core-collapse supernovae evolve from such massive progenitors that they lose their hydrogen envelopes via stellar winds (Hint: integrate the IMF)? The observed fraction is N(SN$_{Ib/c}$)/N(SN$_{CC}$) = 25%. Can massive Wolf-Rayet stars explain the majority of SN Ib/c? If not, what other evolutionary channel is required?**

Use Salpeter IMF for massive stars and compare the fraction of stars with $M > 43.4M_\odot$ to the number of stars with $M > 8M_\odot$. K is a proportionality constant, which cancels.

$$NdM = KM^{-2.35}dM$$
$$\int_x^{120M_\odot} NdM = K\int_x^{120M_\odot} M^{-2.35}dM$$
$$NM\Big|_x^{120M_\odot} = -\frac{K}{1.35}M^{-1.35}\Big|_x^{120M_\odot}$$
$$N\left(120M_\odot - x\right) = -\frac{K}{1.35}\left[120M_\odot^{-1.35} - x^{-1.35}\right]$$
$$N = \frac{K}{1.35\left(120M_\odot - x\right)}\left[x^{-1.35} - 120^{-1.35}\right]$$

$$\boxed{N\left(m > 8M_\odot\right) = 3.89 \times 10^{-4}K}$$
$$\boxed{N\left(m > 43.4M_\odot\right) = 4.45 \times 10^{-5}K}$$

$$\frac{N_{SNIb/c}}{N_{SN_{cc}}} = \frac{N\left(m > 43.4M_\odot\right)}{N\left(m > 8M_\odot\right)}$$
$$\frac{N_{SNIb/c}}{N_{SN_{cc}}} = \frac{4.45 \times 10^{-5}K}{3.89 \times 10^{-4}K}$$

$$\boxed{\frac{N_{SNIb/c}}{N_{SN_{cc}}} = 11.4\%}$$

Using the Salpeter IMF, it was found that Wolf-Rayet stars make up 11.4% of all core-collapse supernovae. This fraction accounts for less than half observed Type Ib/c supernovae. The other 13.6% could arise from progenitors of massive star binaries. As discussed in class, nearly all massive stars are in at least binary systems, and many exist even in tertiary/higher order systems. It is entirely feasable then that a massive star binary contains a star that has evolved off the main sequence, and transfers its mass to its companion. This would result in a core-collapse supernova that lacks H-lines because the outer H-layers of the evolved star were stripped away by the companion as it expands beyond the Roche lobe of the system.
Code:

```
In [24]: N_M1 = (8 **-1.35 -120 **-1.35) /(1.35 *(120 -8))
         # NP Using IMF to calculate stars with M > 8M_o
         N_M2 = (M_wr **-1.35 -120 **-1.35) /(1.35 *(120 -M_wr))
         # NP Using IMF to calculate stars with M > 43.4M_o
         print('N(SNcc) = ' +format(N_M1, '.2E') +' K')
         print('N(SNIb/c) = ' +format(N_M2, '.2E') +' K')
         # NP Printing results
         f = N_M2 /N_M1 *100
         # NP Calculating fraction of SNIb/c to all cc SN
         print('N(SNIb/c) /(N(SNcc)) = ' +format(f, '.2E') +' %')
         # NP Printing percentage

         N(SNcc) = 3.89E-04 K
         N(SNIb/c) = 4.45E-05 K
         N(SNIb/c) /(N(SNcc)) = 1.14E+01 %
```

x 14/15

2. **(15%): Accretion onto compact objects from binary companions. Assuming a fixed mass-transfer rate of $\dot{M} = 10^{-6} M_\odot$ yr^{-1}, determine the accretion luminosity (in $L_\odot$), effective temperature (in K) of the inner edge of the accretion disk, and the corresponding peak wavelength λ (in nm) and energy (in eV or keV) of emission for accretion onto a:**
   Functions used in this problem:

```
In [66]: def L(M_dot, M, R):
             '''Function to calculate L in L_o of an accreting object
             ---------------------------------------------------------
             Inputs:
             M_dot: float. Accretion rate, M_o yr^-1
             M: float. Mass of accreting object, M_o
             R: float. Radius of accreting object, R_o
             ---------------------------------------------------------
             Outputs:
             L: float. Accretion luminosity, L_o'''
             L = G *M *M_o *M_dot *M_o /(np.pi *10 **7) /2 /R /R_o
             return (L /L_o)

         def T(L, R):
             '''Function to calculate T in K of an accreting object
             ---------------------------------------------------------
             Inputs:
             L: float. Accretion luminosity, L_o
             R: float. Radius of accreting object, R_o
             ---------------------------------------------------------
             Outputs:
             T: float. Effective temperature, K'''
             T = ((L * L_o) /(4 *np.pi *o *(R *R_o) **2)) **0.25
             return T

         def peak_l(T):
             '''Function to calculate peak wavelength of a black body
             ---------------------------------------------------------
             Inputs:
             T: float. Effective temperature, K
             ---------------------------------------------------------
             Outputs:
             l: float. Peak wavelength of black body emission, nm'''
             l = 2.898e-3 /T
             return l *1e9

         def energy(l):
             '''Function to calculate energy of photons from accretion
                 ---------------------------------------------------------
                 Inputs:
                 l: float. Peak wavelength of black body emission, nm
                 ---------------------------------------------------------
                 Outputs:
                 E: float. Energy of photon of given wavelength, kev'''
             E = h *c / (l *1e-9) /1.6e-19 /1000
             return E
```

   (a) $0.6M_\odot$ **white dwarf**

$$R \approx 0.01 R_\odot \left(\frac{M}{0.8M_\odot}\right)^{-1/3}$$

$$\boxed{R \approx 0.0110 R_\odot}$$

$$L = \frac{GM\dot{M}}{2R}$$

$$\boxed{L = 8.69 \times 10^2 L_\odot}$$

$$L = 4\pi\sigma R^2 T_{eff}^4$$

use disk equation → factor of ~~two~~ 2 off in T_{eff} Okay

$$T_{eff}^4 = \frac{L}{4\pi\sigma R^2}$$

$$T_{eff} = \left(\frac{L}{4\pi\sigma R^2}\right)^{0.25}$$

$$\boxed{T_{eff} = 2.99 \times 10^5 \text{ K}}$$

$$T_{eff}\lambda_p = 2.898 \times 10^{-3} \text{ mK}$$

$$\lambda_p = \frac{2.898 \times 10^{-3} \text{ mK}}{T_{eff}}$$

$$\boxed{\lambda_p = 9.70 \text{ nm}}$$

$$E = \frac{hc}{\lambda}$$

$$\boxed{E = 0.128 \text{ keV}}$$

Code:

```
In [17]: M_wd = 0.6
         # NP Mass of white dwarf in solar masses
         R_wd = 0.01 *(M_wd /0.8) **(-1 /3)
         # NP Radius of white dwarf in solar radii
         print('WD radius: ' +format(R_wd, '.2E') +' R_o')
         L_wd = L(10 **-6, M_wd, R_wd)
         # NP Luminosity of white dwatf in solar luminosities
         print('Accretion luminosity: ' +format(L_wd, '.2E') +' L_o')
         # NP Printing result
         T_wd = T(L_wd, R_wd)
         # NP Calculating effective temperature of accretion for white dwarf
         print('Effective temperature: ' +format(T_wd, '.2E') +' K')
         # NP Printing result
         l_wd = peak_l(T_wd)
         # NP Calculating peak wavelength of emission for accreting white dwarf
         print('Peak wavelength: ' +format(l_wd, '.2E') +' nm')
         # NP Printing result
         E_wd = energy(l_wd)
         print('Photon energy: ' +format(E_wd, '.2E') +' keV')

         WD radius: 1.10E-02 R_o
         Accretion luminosity: 8.69E+02 L_o
         Effective temperature: 2.99E+05 K
         Peak wavelength: 9.70E+00 nm
         Photon energy: 1.28E-01 keV
```

(b) $2M_\odot$ **neutron star**

$$R \approx 10 \text{ km}$$

$$L = \frac{GM\dot{M}}{2R}$$

$$\boxed{L = 2.22 \times 10^6 L_\odot}$$

$$T_{eff} = \left(\frac{L}{4\pi\sigma R^2}\right)^{0.25}$$

$$\boxed{T_{eff} = 5.88 \times 10^7 \text{ K}}$$

$$\lambda_p = \frac{2.898 \times 10^{-3} \text{ mK}}{T_{eff}}$$

$$\boxed{\lambda_p = 4.93 \times 10^{-2} \text{ nm}}$$

$$E = \frac{hc}{\lambda}$$

$$\boxed{E = 25.2 \text{ keV}}$$

Code:

```
In [18]: M_ns = 2
         # NP Mass of neutron star in solar masses
         R_ns = 10000 /R_o
         # NP Radius of Netron star in solar radii
         L_ns = L(10 **-6, M_ns, R_ns)
         # NP Luminosity of neutron star in solar luminosities
         print('Accretion luminosity: ' +format(L_ns, '.2E') +' L_o')
         # NP Printing result
         T_ns = T(L_ns, R_ns)
         # NP Calculating effective temperature of accretion for neutron star
         print('Effective temperature: ' +format(T_ns, '.2E') +' K')
         # NP Printing result
         l_ns = peak_l(T_ns)
         # NP Calculating peak wavelength of emission for accreting neutron star
         print('Peak wavelength: ' +format(l_ns, '.2E') +' nm')
         # NP Printing result
         E_ns = energy(l_ns)
         print('Photon energy: ' +format(E_ns, '.2E') +' keV')

         Accretion luminosity: 2.22E+06 L_o
         Effective temperature: 5.88E+07 K
         Peak wavelength: 4.93E-02 nm
         Photon energy: 2.52E+01 keV
```

(c) $5M_\odot$ black hole

$$R = \frac{2GM}{c^2}$$

$$\boxed{R = 2.13 \times 10^{-5} R_\odot}$$

$$L = \frac{GM\dot{M}}{2R}$$

$$\boxed{L = 3.74 \times 10^6 L_\odot}$$

$$T_{eff} = \left(\frac{L}{4\pi\sigma R^2}\right)^{0.25}$$

$$\boxed{T_{eff} = 5.50 \times 10^7 \text{ K}}$$

$$\lambda_p = \frac{2.898 \times 10^{-3} \text{ mK}}{T_{eff}}$$

$$\boxed{\lambda_p = 5.27 \times 10^{-2} \text{ nm}}$$

$$E = \frac{hc}{\lambda}$$

$$\boxed{E = 23.6 \text{ keV}}$$

Code:

```
In [29]: M_bh = 5
         # NP Mass of black hole in solar masses
         R_bh = 2 *G *M_bh *M_o /(c **2) /R_o
         # NP Schwarzschild radius of black hole in solar radii
         print('Radius of black hole: ' +format(R_bh, '.2E') +' solar radii')
         # NP Printing result
         L_bh = L(10 **-6, M_bh, R_bh)
         # NP Luminosity of black hole in solar luminosities
         print('Accretion luminosity: ' +format(L_bh, '.2E') +' L_o')
         # NP Printing result
         T_bh = T(L_bh, R_bh)
         # NP Calculating effective temperature of accretion for black hole
         print('Effective temperature: ' +format(T_bh, '.2E') +' K')
         # NP Printing result
         l_bh = peak_l(T_bh)
         # NP Calculating peak wavelength of emission for accreting black hole
         print('Peak wavelength: ' +format(l_bh, '.2E') +' nm')
         # NP Printing result
         E_bh = energy(l_bh)
         print('Photon energy: ' +format(E_bh, '.2E') +' keV')
         # NP Printing result

         Radius of black hole: 2.13E-05 solar radii
         Accretion luminosity: 3.74E+06 L_o
         Effective temperature: 5.50E+07 K
         Peak wavelength: 5.27E-02 nm
         Photon energy: 2.36E+01 keV
```

+35/35

3. **(35%): Energetics of Type Ia Supernovae:**

(a) **Calculate the energy (in erg) produced from the thermonuclear explosion of a Chandrasekhar-mass white dwarf. For simplicity, assume the WD was pure O-16 and all of its mass was fused into Ni-56.**

The process of going from ^{16}O to ^{56}Ni takes on average 3.5 ^{16}O for every ^{56}Ni.

$$E = fMc^2$$

$$E = \left(\frac{3.5\ ^{16}\mathrm{O} - ^{56}\mathrm{Ni}}{^{56}\mathrm{Ni}}\right)(1.4M_\odot)\,c^2$$

$$^{16}\mathrm{O} = 15.99491461956 \text{ Da}$$

$$^{56}\mathrm{Ni} = 55.942128 \text{ Da}$$

$$\boxed{E = 1.81 \times 10^{51} \text{erg}}$$

Code:

```
In [36]: m_o16 = 15.99491461956
         # NP Mass of 160 in Daltons
         m_ni56 = 55.942128
         # NP Mass of 56Ni in Daltons
         E = ((3.5 *15.99491461956) -55.942128) /(55.942128) *1.4 *M_o *c **2
         # NP Energy produced from the conversion of 1.4 solar masses of 160 to 56Ni
         print('Energy from Type I SN = ' +format(E *1e7, '.2E') +' ergs')
         # NP Printing result in ergs

         Energy from Type I SN = 1.81E+51 ergs
```

(b) **A normal SN Ia near the middle of the Phillips relation peaks at $M_{bol} = -19.3$ mag (they're standard candles!) for about 20 days. Compute the peak luminosity (in $L_\odot$) and total radiated energy (in erg) in photons.**

Since we are working in absolute magnitudes, we can work in luminosity rather than flux, because luminosity is related to the flux. In other words, absolute magnitude is a measure of the flux at 10 pc, so we can ignore the pre-factor of $1/4\pi r^2$.

$$M_2 - M_1 = -2.5 \log\left[\frac{L_2}{L_1}\right]$$

$$\log\left[\frac{L_2}{L_1}\right] = \frac{M_1 - M_2}{2.5}$$

$$\frac{L_2}{L_1} = 10^{\frac{M_1-M_2}{2.5}}$$

$$L_2 = L_1 10^{\frac{M_1-M_2}{2.5}}$$

$$L_1 = 1L_\odot$$

$$M_1 = 4.75$$

$$M_2 = -19.3$$

$$\boxed{L_2 = 4.17 \times 10^9 L_\odot}$$

$$E = L\Delta t$$

$$\Delta t = 20 \text{ days}$$

$$\boxed{E = 2.76 \times 10^{49} \text{ erg}}$$

Code:

```
In [45]: M_bolsn = -19.3
         # NP Bolometric magnitude of SN
         M_bolo = 4.75
         # NP Bolometric magnitude of Sun
         L_sn = 10 **((M_bolo -M_bolsn) /2.5)
         # NP Calculating luminosity of supernova in L sun
         print('Supernova luminosity: ' +format(L_sn, '.2E') +' L_o')
         # NP Printing result
         E_sn = L_sn *20 *24 *60 *60 *L_o *1e7
         # NP Calculating energy of supernova in ergs
         print('Energy of supernova in photons: ' +format(E_sn, '.2E') +' ergs')
         # NP Printing result

         Supernova luminosity: 4.17E+09 L_o
         Energy of supernova in photons: 2.76E+49 ergs
```

(c) **Observed SN Ia ejecta speeds are 10,000 km s^{-1}. What is the kinetic energy of an SN Ia?**

Assuming entire mass of neutron star is dispersed at uniform speed.

$$E_k = 1/2mv_{ej}^2$$

$$m = 1.4M_\odot$$

$$v_{ej} = 10000 \text{ km s}^{-1}$$

$$\boxed{E_k = 1.40 \times 10^{51}}$$

Code:

```
In [53]: E_k = 0.5 *1.4 *M_o *(10000 *1000) **2 *1e7
         # NP Calculating kinetic energy in ejecta in ergs
         print('Kinetic energy of ejecta: ' +format(E_k, '.2E') +' ergs')
         # NP Printing result

         Kinetic energy of ejecta: 1.40E+51 ergs
```

(d) **Each nuclear reaction of fusing O into Ni produces a neutrino with an energy of 4 MeV. What is the total energy (in erg) produced in neutrinos?**

Find the energy produced by one nuclear reaction of taking 3.5 atoms of ^{16}O and producing one atom of ^{56}Ni. Then compare to the total energy produced when $1.4M_\odot$ of ^{16}O is converted to ^{56}Ni to find the number of reactions took place.

$$E = \Delta mc^2$$
$$E = (3.5\ ^{16}O - ^{56}\mathrm{Ni})\, c^2$$
$$E_{tot} = \left(\frac{3.5\ ^{16}\mathrm{O} - ^{56}\mathrm{Ni}}{^{56}\mathrm{Ni}}\right)(1.4M_\odot)\, c^2$$
$$NE = E_{tot}$$
$$N = \frac{E_{tot}}{E}$$
$$N = \frac{\left(\frac{3.5\ ^{16}\mathrm{O} - ^{56}\mathrm{Ni}}{^{56}\mathrm{Ni}}\right)(1.4M_\odot)\, c^2}{(3.5\ ^{16}O - ^{56}\mathrm{Ni})\, c^2}$$
$$N = \frac{1.4M_\odot}{^{56}\mathrm{Ni}}$$
$$E_\nu = N\,(4\mathrm{Mev/reaction})$$
$$\boxed{E_\nu = 1.93 \times 10^{50}\ \mathrm{erg}}$$

Code:

```
In [60]: N = 1.4 *M_o /(1.66053906660e-27 *m_ni56)
         # NP Number of nuclear reactions to turn 1.4 M_o of O16 to Ni56
         E_v = N * 4 *1e6 *1.6e-19 *1e7
         # NP Energy in neutrinos in ergs assuming each reaction produces 4Mev in
         # NP neutrinos
         print('Energy in neutrinos: ' +format(E_v, '.2E') +' ergs')

         Energy in neutrinos: 1.93E+50 ergs
```

(e) **Each Ni-56 atom subsequently beta decays to Co-56 (half-life of 6.08 days) and then beta decays again to Fe-56 on longer timescales (half-life of 77.3 days). What is the total energy (in erg) from radioactive decay?**

$$\Delta E_1 = \frac{^{56}\mathrm{Ni} - ^{56}\mathrm{Co}}{^{56}\mathrm{Co}} 1.4M_\odot c^2$$
$$\Delta E_2 = \frac{^{56}\mathrm{Co} - ^{56}\mathrm{Fe}}{^{56}\mathrm{Fe}} 1.4M_\odot c^2$$
$$\Delta E_{rad} = \Delta E_1 + \Delta E_2$$
$$\Delta E_{rad} = \left(\frac{^{56}\mathrm{Ni} - ^{56}\mathrm{Co}}{^{56}\mathrm{Co}} + \frac{^{56}\mathrm{Co} - ^{56}\mathrm{Fe}}{^{56}\mathrm{Fe}}\right) 1.4M_\odot c^2$$
$$\boxed{\Delta E_{rad} = 3.24 \times 10^{50}\ \mathrm{erg}}$$

Code:

```
In [67]: m_co56 = 55.939838
         # NP Mass of Co56 in Daltons
         m_fe56 = 55.934936
         # NP Mass of Fe56 in Daltons
         E_rad = (((m_ni56-m_co56) /(m_co56)) +((m_co56-m_fe56) /(m_fe56))) *1.4\
             *M_o *c **2 *1e7
         # NP Calculating energy in radiactive decays in ergs
         print('Energy from radiactive decays: ' +format(E_rad, '.2E') +' ergs')
         # NP Printing result

         Energy from radiactive decays: 3.24E+50 ergs
```

(f) **You should find that the radiated, kinetic, neutrino, and radioactive energies roughly sum to the total energy produced by nucleosynthesis in part a. But one energy proces constitutes**

the majority, two are $\sim 10\%$, and the other is only $\sim 1\%$. Which is which?

$$\Delta E = \frac{E_p + E_k + E_\nu + E_{rad}}{E_{SN}}$$

$$\boxed{\Delta E = 1.08}$$

Energies of all of these processes roughly add up to the total energy released from a Type Ia supernova! As shown above, the kinetic energy of the supernova ejecta is the same order of magnitude of the total energy released in a Type Ia SN. The energy in radioactive decays and energy in neutrinos each make up roughly 10% of the total energy of a Type Ia SN. Finally, the energy in photons accounts for roughly 1% of the total energy released in a Type Ia supernova.
Code:

```
In [74]: DE = (E_sn +E_k +E_v +E_rad) /(E)
         # NP Ratio of sum of energies and total predicted Type Ia SN energy
         print('Delta E: ' +format(DE, '.2E'))
         # NP Printing result

         Delta E: 1.08E+00
```

+20/20 great!

4. **(20%): Supernova Remnant (SNR):**

(a) **The initial SN shockwave expands at nearly constant velocity v_{sh} until the swept-up mass of the ISM is comparable to the initial ejecta mass, at which point the SNR transitions into an adiabatic Sedov-Taylor expansion. Assuming an SN ejecta mass of $M_{ej} = 5M_\odot$, SN kinetic energy of $E_{kin} = 10^{51}$ erg, and typical ISM density of $n = 1$ cm^{-3}, compute the shock velocity v_{sh} (in km s^{-1}), radius r_{sh} (in pc), and age t_{sh} (in yr) of this transition.**

$$\rho_{ISM} = \mu m_H n$$
$$\mu \approx 1.4$$
$$\rho_{ISM} = 2.34 \times 10^{-24}\ \text{g cm}^{-3}$$
$$E_k = 10^{51}\ \text{erg}$$
$$E_k = \frac{1}{2} M_{ej} v_{sh}^2$$
$$v_{sh} = \left(\frac{2E_k}{M_{ej}}\right)^{0.5}$$
$$M_{ej} = \frac{4}{3}\pi r_{sh}^3 \rho_{ISM}$$
$$r_{sh} = \left(\frac{3M_{ej}}{4\pi\rho_{ISM}}\right)^{1/3}$$
$$t_{sh} = \frac{r_{sh}}{v_{sh}}$$

$$\boxed{v_{sh} = 4470\ \text{km/s}}$$
$$\boxed{r_{sh} = 3.26\ \text{pc}}$$
$$\boxed{t_{sh} = 717\ \text{years}}$$

Code:

```
In [137]: rho_ISM = 1.4 *m_H *1000
          # NP Calculating ISM density assuming mu = 1.4 in gcm^-3
          print('ISM density: ' +format(rho_ISM, '.2E') +' g cm^-3')
          # NP Printing result
          E_k = 1e51 *1e-7
          # Kinetic energy in ergs
          M_ej = 5 *M_o
          # NP Mass of SN ejecta
          v = np.sqrt((E_k *2) /(M_ej)) /1000
          # NP Speed of shock in km/s
          print('Shock speed: ' +format(v, '.2E') +' km/s')
          # NP Printing result
          r_sh = ((3 *M_ej) /(4 *np.pi *rho_ISM *1000)) **(1/3) /(3.09e16)
          # NP Calculating shock radius in pc
          print('Shock radius: ' +format(r_sh, '.2E') +' pc')
          # NP Printing result
          t_sh = (r_sh) /(v) /(np.pi *1e7) *3.09e13
          # NP Calculating shock age in years
          print('Shock age: ' +format(t_sh, '.2E') +'years')
          # NP Printing result

          ISM density: 2.34E-24 g cm^-3
          Shock speed: 4.47E+03 km/s
          Shock radius: 3.26E+00 pc
          Shock age: 7.17E+02years
```

(b) **The adiabatic Sedov-Taylor phase transitions into a slower momentum-conserving snowplow phase when radiative losses are comparable to the initial SN kinetic energy. Assuming the SNR radiates a cumulative energy of $E_{rad} = n_{ISM}^2 \Lambda \left(4\pi r_{sh}^3/3\right) t_{sh}$, where the gas cooling function is $\Lambda = 10^{-21}$ erg cm^3 s^{-1}, estimate the age t_{sh} and corresponding radius r_{sh} of the SN shockwave near this transition. What is the total accumulated shocked mass of the SNR, and how many**

times the ejecta mass is this?

$$E_{rad} = n_{ISM}^2 \Lambda \left(\frac{4}{3}\pi r_{sh}^3\right) t_{sh}$$

$$r_{sh} = 2.3 \text{ pc} \left(\frac{\rho_{ISM}}{10^{-24} \text{ g cm}^{-3}}\right)^{-1/5} \left(\frac{t_{sh}}{100 \text{ yrs}}\right)^{2/5}$$

$$E_{rad} = n_{ISM}^2 \Lambda \left(\frac{4}{3}\pi \left(2.3 \text{ pc} \left(\frac{\rho_{ISM}}{10^{-24} \text{ g cm}^{-3}}\right)^{-1/5} \left(\frac{t_{sh}}{100 \text{ yrs}}\right)^{2/5}\right)^3\right) t_{sh}$$

$$E_{rad} = n_{ISM}^2 \Lambda \left(\frac{4}{3}\pi \left(2.3 \text{ pc}\right)^3 \left(\frac{\rho_{ISM}}{10^{-24} \text{ g cm}^{-3}}\right)^{-3/5} \left(\frac{t_{sh}}{100 \text{ yrs}}\right)^{6/5}\right) t_{sh}$$

$$t_{sh} = \frac{E_{rad}}{n_{ISM}^2 \Lambda \left(\frac{4}{3}\pi \left(2.3 \text{ pc}\right)^3 \left(\frac{\rho_{ISM}}{10^{-24} \text{ g cm}^{-3}}\right)^{-3/5}\right)} \left(\frac{t_{sh}}{100 \text{ yrs}}\right)^{-6/5}$$

$$t_{sh} \left(\frac{t_{sh}}{100 \text{ yrs}}\right)^{6/5} = \frac{E_{rad}}{n_{ISM}^2 \Lambda \left(\frac{4}{3}\pi \left(2.3 \text{ pc}\right)^3 \left(\frac{\rho_{ISM}}{10^{-24} \text{ g cm}^{-3}}\right)^{-3/5}\right)}$$

$$\left(t_{sh}\right)^{11/5} \left(100 \text{ yrs}\right)^{-6/5} = \frac{E_{rad}}{n_{ISM}^2 \Lambda \left(\frac{4}{3}\pi \left(2.3 \text{ pc}\right)^3 \left(\frac{\rho_{ISM}}{10^{-24} \text{ g cm}^{-3}}\right)^{-3/5}\right)}$$

$$\left(t_{sh}\right)^{11/5} = \left(100 \text{ yrs}\right)^{6/5} \frac{E_{rad}}{n_{ISM}^2 \Lambda \left(\frac{4}{3}\pi \left(2.3 \text{ pc}\right)^3 \left(\frac{\rho_{ISM}}{10^{-24} \text{ g cm}^{-3}}\right)^{-3/5}\right)}$$

$$t_{sh} = \left(100 \text{ yrs}\right)^{6/11} \left(\frac{E_{rad}}{n_{ISM}^2 \Lambda \left(\frac{4}{3}\pi \left(2.3 \text{ pc}\right)^3 \left(\frac{\rho_{ISM}}{10^{-24} \text{ g cm}^{-3}}\right)^{-3/5}\right)}\right)^{5/11}$$

$$r_{sh} = 2.3 \text{ pc} \left(\frac{\rho_{ISM}}{10^{-24} \text{ g cm}^{-3}}\right)^{-1/5} \left(\frac{t_{sh}}{100 \text{ yrs}}\right)^{2/5}$$

$$v_{sh} = 9000 \text{ km s}^{-1} \left(\frac{\rho_{ISM}}{10^{-24} \text{ g cm}^{-3}}\right)^{-1/5} \left(\frac{t_{sh}}{100 \text{ yrs}}\right)^{-3/5}$$

$$E_k = \frac{1}{2} M_{SNR} v_{sh}^2$$

$$M_{SNR} = \frac{2E_k}{v_{sh}^2}$$

$$M_{SNR} = \frac{2E_k}{\left(9000 \text{ km s}^{-1}\right)^2 \left(\frac{\rho_{ISM}}{10^{-24} \text{ g cm}^{-3}}\right)^{-2/5} \left(\frac{t_{sh}}{100 \text{ yrs}}\right)^{-6/5}}$$

$$\boxed{t_{sh} = 3.32 \times 10^4 \text{ yrs}}$$

$$\boxed{r_{sh} = 19.8 \text{ pc}}$$

$$\boxed{M_{SNR} = 1.84 \times 10^3 M_\odot}$$

The accumulated shocked mass is approximately 10^3 times greater!
Code:

```
In [151]: t_sh2 = (100 *np.pi *1e7) **(6/11) *((3 *E_k *1e7) /(1e-21 *4 *np.pi\
              *(2.3 *3.09e18) **3 *(rho_ISM *1e24) **(-3/5))) **(5/11)\
              /(np.pi *1e7)
          # NP Calculating when momentum conserving phase occurs in years
          print('Momentum conserving phase occurs at age: ' +format(t_sh2, '.2E')\
              +' years')
          # NP Printing result
          r_sh2 = 2.3 *((rho_ISM) /(1e-24)) **(-1/5) *((t_sh2) /(100)) **(2/5)
          print('Shock radius at momentum phase: ' +format(r_sh2, '.2E')\
              +' pc')
          # NP Printing result
          M_SNR = (2 *E_k) /((9000 *1000) **2 *(rho_ISM *1e24) **(-2/5)\
              *(t_sh2/100) **(-6/5)) /M_o
          # NP Calculating mass of SNR in Solar masses
          print('SNR Mass : ' +format(M_SNR, '.2E') +' M_o')
          # NP Printing result

          Momentum conserving phase occurs at age: 3.32E+04 years
          Shock radius at momentum phase: 1.98E+01 pc
          SNR Mass : 1.84E+03 M_o
```

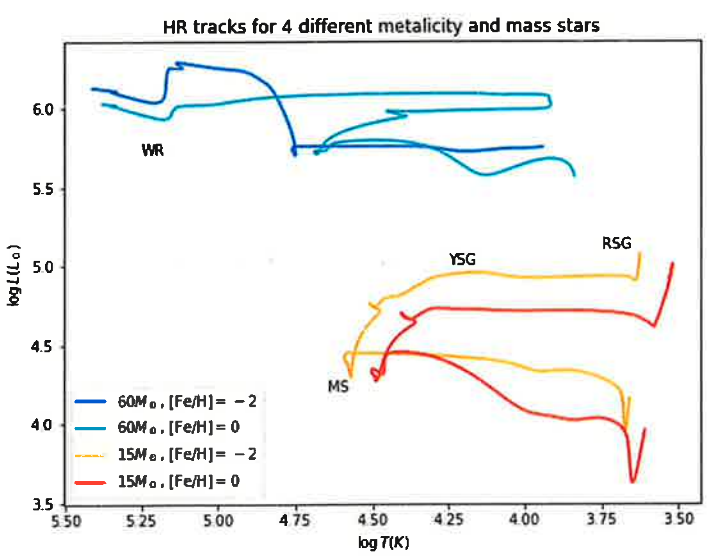
+15/15

5. **(15%): Download from WyoCourses the MESA evolutionary tracks for $15M_\odot$ and $60M_\odot$ stars at both $[Fe/H] = -2$ and 0 metallicities. The tables include 77 columns (mostly surface abundances), but all you will need is stellar age (in yr), $\log L/L_\odot$, and $\log T_{eff}/K$.**

Reading in tables:

```
In [158]: L1, T1, R1, ages1 = np.loadtxt('/d/users/nikhil/Downloads/MESA_60p0'
              'Msun_FeHm2p0.txt', usecols = (6, 11, 13, 0), unpack = True)
          L2, T2, R2, ages2 = np.loadtxt('/d/users/nikhil/Downloads/MESA_60p0'
              'Msun_FeH0p0.txt', usecols = (6, 11, 13, 0), unpack = True)
          L3, T3, R3, ages3 = np.loadtxt('/d/users/nikhil/Downloads/MESA_15p0'
              'Msun_FeHm2p0.txt', usecols = (6, 11, 13, 0), unpack = True)
          L4, T4, R4, ages4 = np.loadtxt('/d/users/nikhil/Downloads/MESA_15p0'
              'Msun_FeH0p0.txt', usecols = (6, 11, 13, 0), unpack = True)
          # NP Reading in data for different mass stars with different
          # NP metallicities
```

(a) **Plot the four tracks on the same HR diagram. Label the regions corresponding to the MS, YSG, RSG, and WR.**



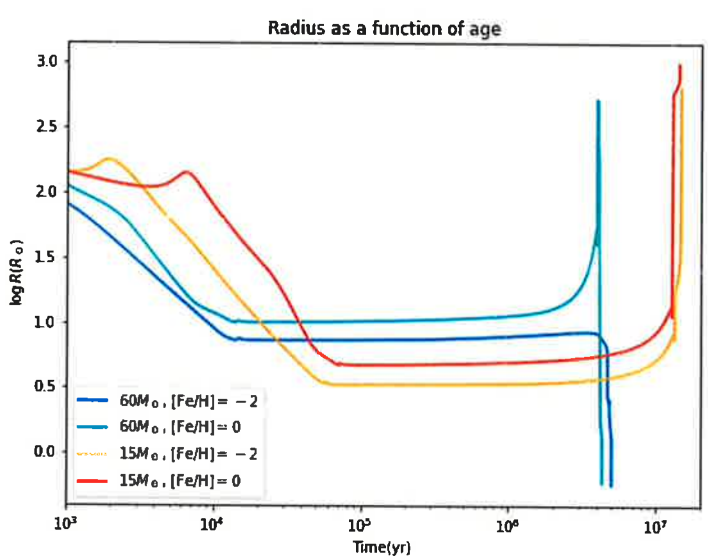
Code:

```
In [169]: plt.figure(figsize = [8, 6])
          # NP Making figure larger
          plt.plot(T1, L1, 'b', label = r'$60M_\odot$, [Fe/H]$=-2$')
          plt.plot(T2, L2, 'c', label = r'$60M_\odot$, [Fe/H]$=0$')
          plt.plot(T3, L3, color = 'orange', label = r'$15M_\odot$,'
              ' [Fe/H]$=-2$')
          plt.plot(T4, L4, 'r', label = r'$15M_\odot$, [Fe/H]$=0$')
          # NP Plotting temperaures and luminosities
          plt.text(4.65, 4.2, 'MS')
          plt.text(4.25, 5.0, 'YSG')
          plt.text(3.75, 5.1, 'RSG')
          plt.text(5.25, 5.7, 'WR')
          # NP Labeling different phases in stellar evolution
          plt.gca().invert_xaxis()
          # NP Flipping x-axis
          plt.legend()
          # NP Making legend
          plt.xlabel(r'$\log T(K)$')
          plt.ylabel(r'$\log L(L_\odot)$')
          # NP Labeling axes
          plt.title('HR tracks for 4 different metalicity and mass stars')
          # NP Labeling plot
          plt.savefig('/d/www/nikhil/public_html/ASTR5420/images/HRtracksM.png')
          # NP Saving figure
```

(b) **Describe the differences between the metal-poor and solar-metallicity tracks. At what phase of evolution do they differ the most? Explain what causes this difference.**

Generally, the metal-poor and solar metallicity tracks differ in that the metal-poor stars are hotter and more luminous. The greatest difference between two tracks of the same mass are shown in the paths taken by the $60M_{\odot}$ stars. Both stars are about the same luminosity and temperature, with the metal-poor star being slightly hotter and more luminous, but the metal-poor star jumps off the MS and becomes a WR star, while the solar-metallicity star first evolves into a YSG and RSG before it becomes a Wolf-Rayet star. This can be explained by the greater mass-loss rates of the solar-metallicity star during its MS lifetime. The solar-metallicity star loses mass during it's lifetime, by metal-driven winds, which in-turn decreases its mass by the time it ends its main sequence. The solar-metallicity massive star therefore loses it's outer layers before the core collapses into a supernova, resulting in the star evolving into a YSG and RSG before becoming a Wolf-Rayet star.

(c) **Separately plot the stellar radii R (in $R_{\odot}$) as a function of time. Adjust the time scale to feature the different phases of evolution.**



Code:

```
In [187]: plt.figure(figsize = [8, 6])
          # NP Making figure larger
          plt.plot(ages1, R1, 'b', label = r'$60M_\odot$, [Fe/H]$=-2$')
          plt.plot(ages2, R2, 'c', label = r'$60M_\odot$, [Fe/H]$=0$')
          plt.plot(ages3, R3, color = 'orange', label\
              = r'$15M_\odot$, [Fe/H]$=-2$')
          plt.plot(ages4, R4, 'r', label = r'$15M_\odot$, [Fe/H]$=0$')
          # NP Plotting radius as a function of age for all stars
          plt.legend()
          # NP Creating legend
          plt.xlabel(r'Time(yr)')
          plt.ylabel(r'$\log R(R_\odot)$')
          plt.title(r'Radius as a function of age')
          # NP Labeling figure
          plt.xscale('log')
          # NP Scaling x-axis to log scale
          plt.xlim(1 *10 **3, 2*10**7)
          plt.savefig('/d/www/nikhil/public_html/ASTR5420/images/Radiustime'
              'stars2.png')
          # NP Saving figure
```

243 (d) **Report the maximum radii (in $R_\odot$) for all four tracks.**

	Star	$\log R_{max}$	
	$60M_\odot\ [Fe/H] = -2$	2.52	$\sim 300\ R_\odot$ ✓
244	$60M_\odot\ [Fe/H] = 0$	2.71	
	$15M_\odot\ [Fe/H] = -2$	2.81	
	$15M_\odot\ [Fe/H] = 0$	2.99	$\sim 1000\ R_\odot$ ✓

245 Code:

```
In [188]: print('60M_o Z= -2 log[radius]: ' +format(np.max(R1),'.2E'))
          print('60M_o Z= 0 log[radius]: ' +format(np.max(R2),'.2E'))
          print('15M_o Z= -2 log[radius]: ' +format(np.max(R3),'.2E'))
          print('15M_o Z= 0 log[radius]: ' +format(np.max(R4),'.2E'))

          60M_o Z= -2 log[radius]: 2.52E+00
          60M_o Z= 0 log[radius]: 2.71E+00
          15M_o Z= -2 log[radius]: 2.81E+00
          15M_o Z= 0 log[radius]: 2.99E+00
```